



**Escola Politècnica Superior  
d'Enginyeria de Vilanova i la Geltrú**

UNIVERSITAT POLITÈCNICA DE CATALUNYA

# TREBALL FINAL DE GRAU

**TÍTOL:** Desenvolupament d'un mòdul per a l'aplicació Android d'ajut en la comunicació i l'expressió per a nens autistes 'ECO'

**AUTORS:** Castells Vivó, Alfred

**DATA DE PRESENTACIÓ:** Febrer, 2016

**COGNOMS:** Castells Vivó

**NOM:** Alfred

**TITULACIÓ:** Grau en Enginyeria Informàtica

**PLA:** 2010

**DIRECTOR:** Josep Maria Merenciano Saladrígues

**DEPARTAMENT:** Departament de Ciències de la Computació

**QUALIFICACIÓ DEL TFG**

**TRIBUNAL**

**PRESIDENT**

**SECRETARI**

**VOCAL**

**DATA DE LECTURA:**

**Aquest Projecte té en compte aspectes mediambientals:** ☐ **Sí** ☐ **No**

## RESUM

Amb aquest treball es pretén construir un nou mòdul extensiu per a l'aplicació d'ajut en la comunicació dels autistes 'ECO', desenvolupada per professors de la UPC (Universitat Politècnica de Catalunya).

L'aplicació ofereix un conjunt d'elements (missatges), disposats en forma de graella i classificats per categories, que representen objectes, idees o emocions. Aquets missatges estan formats per una imatge (principalment) i un vídeo i un so (opcionalment), amb l'objectiu que el nostres usuaris els reconeguin com a elements de la seva vida quotidiana mitjançant la seva reproducció.

Es demana que, amb la incorporació del nostre mòdul, a més a més, l'usuari tingui la possibilitat de construir 'frases', és a dir, conjunts de missatges. També es demana també que, un cop construïda aquesta frase, hi hagi la possibilitat de reproduir-la, acció que ha de ser una reproducció seqüencial de tots els missatges que conformen aquesta frase.

La premissa destacada a tenir en compte, donat el tipus d'usuari al qual va dirigida l'aplicació, és el caràcter intuïtiu que ha d'oferir, en tot moment, aquest nou mòdul. Els nens autistes tenen els sentits de la percepció, intuïció i correlació de conceptes desenvolupats de forma diferent, de manera que tota la interfície, i en especial, les interaccions que oferim, han de ser tant intuïtives com sigui possible.

Així doncs, afegim un nou element a la pantalla corresponent de l'aplicació en forma de llistat vertical, al qual, per mitjà de l'arrossegament dels missatges de la graella, s'hi poden anar afegint missatges. Evidentment, també se'n poden esborrar, acció per la qual es duu a terme amb la mateixa interacció; l'arrossegament, en aquest cas, a una paperera.

Incorporem també un nou botó per a la reproducció de la frase, així com un altre per a poder sortir-ne en qualsevol moment. A més a més, per donar suport visual i fer que tot sigui més intuïtiu, incorporem diverses animacions que ajudaran a comprendre tot el que succeeixi en l'aplicació en tot moment.

Amb la realització d'aquest projecte, he ampliat considerablement els meus coneixement en el camp del desenvolupament Android, i he comprès la feina que suposa incorporar una part nova a un projecte ja construït, com a aspectes destacats de les conclusions a les que arribo.

### Paraules clau (màxim 10):

Interfície	Interaccions	Mòdul	Missatge
Reproducció	Fragment	Graella	Llista
Frase	Animació		

## ABSTRACT

What we want to do in this project is to build a new extensive module for 'ECO', an application developed by UPC (Universitat Politècnica de Catalunya) professors that helps in the autistic kids communication.

The application offers a set of elements (messages), displayed in a grid and classified into categories, which represent objects, ideas or emotions. These messages are composed of an image (mainly) plus a video and a sound (optionally), in order that our users recognize them as elements of their daily life through their reproduction.

I'm asked to, with the incorporation of our module, that the users have the possibility of building 'sentences', that is, sets of messages. I'm also asked to, once built this sentence, that there is also the possibility of reproduce it, an action that should be a sequential reproduction of all the messages that compose this sentence.

The most important premise to consider, given the type of user to which is addressed this application, is the intuitive nature that has to offer, always, this new module. The autistic kids have senses of perception, intuition and concepts correlation developed differently, so that the entire interface, and specially, the interactions that we offer, have to be as intuitive as possible.

So, we add a new element to the appropriate application screen in a vertical list, which, by dragging messages from the grid, these are added into the sentence. Obviously, they can also be deleted by doing the same interaction, dragging, in this case, into a bin.

We also include a new button for the sentence reproduction, and another one to be able to get out of it at any time too. In addition, in order to give visual support and make everything more intuitive, we also incorporate several animations that will help to understand everything that happens in the application.

By doing this project, I've expanded my knowledge in the field of Android development, and I've appreciated the work involved in incorporating a new part to an already built project, as highlights of the conclusions I've arrived.

### Keywords (10 maximum):

Interface	Interactions	Module	Messages
Reproduction	Fragment	Grid	List
Sentence	Animation		



# ÍNDEX

<b>1.- INTRODUCCIÓ</b>	1
1.1.- Motivació	1
1.2.- La idea	1
1.3.- Contextualització	2
1.4.- Objectius	3
1.4.1.- <i>Propòsit general de l'aplicació ECO</i>	3
1.4.2.- <i>Objectius del mòdul propi</i>	4
1.5.- Contingut de la memòria	5
<b>2.- AVALUACIÓ TECNOLÒGICA</b>	6
2.1.- Sistemes operatius	6
2.2.- Llenguatges de programació	7
2.3.- Sistemes Gestors de Bases de Dades (SGBD)	8
2.4.- Programari utilitzat	8
2.4.1.- L'IDE	8
2.4.2.- L'emulador	9
2.4.3.- Dispositius físics	9
<b>3.- PLANIFICACIÓ I DESENVOLUPAMENT</b>	11
3.1.- Històric de desenvolupament	11
3.2.- Work Packages	12
3.3.- Diagrama de <i>Gantt</i>	13
3.4.- Desenvolupament i coordinació	15
3.5.- Metodologies de desenvolupament	16
<b>4.- ANÀLISI DE REQUERIMENTS</b>	17
4.1.- Escenari inicial	17
4.2.- Noves funcionalitats	19
<b>5.- ESPECIFICACIÓ</b>	22
5.1.- Escenari inicial	22
5.1.1.- <i>Diagrama conceptual</i>	22
5.1.2.- <i>Glossari</i>	22
5.1.3.- <i>Restriccions i suposicions semàntiques</i>	23
5.1.4.- <i>Casos d'Ús</i>	23
5.2.- Nova especificació	24
5.2.1.- <i>Nou diagrama conceptual</i>	24
5.2.2.- <i>Ampliació del glossari</i>	24

5.2.3.- <i>Restriccions i suposicions semàntiques</i> .....	25
5.2.4.- <i>Casos d'Ús</i> .....	25
<b>6.- DISSENY</b> .....	27
6.1.- Diagrames de col·laboració.....	27
6.1.1.- <i>Cas d'ús 1: Reproduir missatge</i> .....	27
6.1.2.- <i>Cas d'ús 2: Crear i reproduir frase</i> .....	28
6.2.- Model de components .....	29
6.3.- Disseny de la interfície .....	29
6.3.1.- <i>Disseny dels components</i> .....	30
6.3.2.- <i>Disseny de les interaccions</i> .....	33
6.4.- Contextualització de la interfície .....	40
<b>7.- IMPLEMENTACIÓ</b> .....	42
7.1.- Marc contextual .....	42
7.2.- Implementació de la interfície .....	45
7.2.1.- <i>Implementació dels components</i> .....	45
7.2.2.- <i>Implementació de les interaccions</i> .....	48
7.3.- Especificació, disseny i implementació .....	54
<b>8.- CONCLUSIONS I ASSOLIMENT D'OBJECTIUS</b> .....	56
8.1.- Valoració de la planificació i el desenvolupament.....	56
8.2.- Assoliment d'objectius i requeriments.....	56
8.3.- Valoració personal.....	58
<b>9.- TREBALL FUTUR</b> .....	60
9.1.- Immediat .....	60
9.2.- Millores i propostes d'ampliació .....	61
<b>10.- BIBLIOGRAFIA</b> .....	63

# 1.- INTRODUCCIÓ

## 1.1.- Motivació

Al llarg de la carrera he cursat moltes assignatures i assimilat molts conceptes relacionats amb les diferents disciplines compreses en el ventall de tasques o competències que ha de ser capaç de desenvolupar i dominar un enginyer informàtic. Cap, però, m'ha cridat l'atenció com ho ha fet el **desenvolupament d'aplicacions mòbils**.

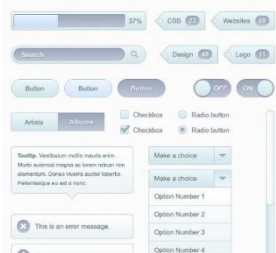
El dels mòbils és un dels camps o terrenys que està experimentant un procés d'expansió més gran en els últims anys, iniciat amb l'emergència dels primers models d'*smartphones* i continuat amb el conseqüent creixement dels sectors relacionats i l'emergència de les *tablets*. I no només parlem d'expansió en quant als diferents dispositius mòbils, sinó també en el desenvolupament dels diferents sistemes operatius i de les aplicacions mòbils, sent aquest últim el concepte cap a on va enfocat el projecte.



Imatge 1: Aplicacions mòbils

Penso que és un món atractiu per l'enginyer informàtic, ja que aquests dispositius suposen un canvi de concepció pel que fa als mòbils, i suposa tot un repte saber trobar els elements per a satisfer les necessitats (manifestes o latents, sobretot aquestes últimes) que puguin tenir els usuaris respecte el que fins fa ben poc era simplement un telèfon i ara és, sens dubte, bastant més.

A més a més, és un sector que ofereix la possibilitat d'aprofundir molt en diferents aspectes a banda dels purament tècnics relatius a la programació del funcionament de l'aplicació en sí.



Imatge 2: Elements interfície

Primer de tot, l'enfocament cap a un públic objectiu, donat que avui dia gairebé tothom disposa d'un *smartphone*, la qual cosa propicia poder diferenciar i seleccionar grups o tipus de persones a qui dirigir l'aplicació. I després, un dels pilars bàsics del desenvolupament mòbil: jugar amb la seva interfície i els elements que la conformen, els gestos, els aspectes i l'adaptació als diferents dispositius sobre la qual volem que s'hi

pugui fer ús.

## 1.2.- La idea

Així doncs, un cop decidit l'àmbit en què desenvolupar el projecte, vaig començar a pensar en algunes idees que poguessin desembocar en alguna aplicació interessant, que representés una eina útil de cara a fer el seguiment o ajudar en algun aspecte quotidià. No obstant, de bon començament ja se'm va presentar una interessant opció proposada per professors de la Universitat Politècnica de Catalunya (UPC) que consistia en crear una



Imatge 3: Idea aplicació



aplicació per a tablets que emulés i ampliés el funcionament d'una aplicació didàctica per a nens autistes en la qual ja portaven temps treballant, l'aplicació **Easy Communicator (ECO)**.

En aquell moment vaig considerar seriosament la proposta i vaig concloure que era un encert treballar en una idea proposada per algú i enfocada una necessitat real, per tal de simular l'escenari on es desenvolupa un producte a petició d'un client, que al cap i a la fi és l'escenari més comú que es pot trobar un enginyer informàtic. Acceptant aquesta proposta, doncs, el projecte també suposava una introducció més propera al món real (el món laboral informàtic).

No obstant, una mica més endavant, a causa de certes circumstàncies, vaig tenir el projecte pausat un temps i al reprendre'l, em va arribar una nova proposta des de la UPC en la qual se m'oferia, en comptes de desenvolupar una aplicació pròpia, treballar en la **mateixa aplicació ECO**, desenvolupant-hi un nou **mòdul/complement** que suposaria una **nova funcionalitat** d'usuari que més endavant detallaré.

En un principi era un pèl reticent amb aquesta idea, ja que tot i no haver avançat molt en la meua pròpia aplicació, sí que tenia coses pensades. També hi era el fet que



*Imatge 4: Incorporació*

aquest canvi suposava incorporar un element o tasca nova al projecte que no havia previst al principi, que era assimilar i comprendre l'estructura i els mètodes de funcionament d'una aplicació "aliena" (i la feina que això suposa) per tal que la meua part s'hi acoblés correctament.

Al final però, una mica en la línia de la reflexió anterior, vaig acabar concloent que podia aportar bastant a la meua formació i a la meua carrera haver treballat un mòdul d'una aplicació més professional i més completa de la que hauria pogut desenvolupar jo sol pel meu compte, tot i perdre el sentit de 'la creació pròpia' del projecte.

D'aquesta manera, el que de bon començament semblava que havia de ser una idea pròpia desenvolupada solament per mi, es va acabar convertint en un **mòdul o bloc nou extensiu** per a una aplicació professional en col·laboració amb la UPC, cosa que fa que el contingut d'aquesta memòria variï una mica de com hauria de ser en un projecte "estàndard".

En els punts restants de la introducció posaré el problema que es vol enfocar en situació i explicaré a grans trets els objectius de l'aplicació i en més detall els del meu mòdul.

### 1.3.- Contextualització

Com comentava abans, tota aplicació està pensada per a un públic objectiu, més o menys gran, a qui anirà dirigida principalment. En el nostre cas, el públic objectiu n'és un de ben particular. L'aplicació ECO està pensada per ser usada pels **nens afectats per l'autisme** i les persones a càrrec d'aquests.

L'autisme és un trastorn psicològic causat bàsicament per problemes en el desenvolupament que afecta a la comunicació, la socialització i la imaginació, entre d'altres aspectes, caracteritzant-se per la intensa concentració d'una persona en el seu propi món interior i la progressiva pèrdua de contacte amb el món real. Els símptomes (poca



*Imatge 5: Autisme*

visió, dificultats en el llenguatge, incomoditat amb el contacte físic) comencen a aparèixer en l'etapa infantil, fent que el nen que els pateix tingui serioses dificultats per expressar-se i comunicar-se.

Així doncs, en l'àmbit que ens interessa, s'ha d'anar amb compte, donat que estem tractant amb un tipus de persones concret, els individus del qual no tenen certs sentits desenvolupats com la resta de les persones. Concretament podríem parlar de la percepció, la intuïció, i també de la correlació de conceptes necessàries per tal d'interactuar amb una aplicació normal de forma òptima.



Imatge 6: Interfície intuïtiva

És important, doncs, enfocar el funcionament i sobretot el disseny de l'aplicació en aquest punt, de manera que la interfície sigui molt **intuïtiva**, fins al punt que a una persona que no pateixi d'autisme li resulti exagerat o innecessari. En aquest sentit cal, més que mai, posar-se en la pell (tot i que sigui complicat) de l'usuari i intentar posar-li les coses

fàcils, simplificant al màxim la distribució i el comportament dels elements de què es componi l'aplicació.

Concloent, el problema al qual la aplicació va enfocada i que té com a objectiu intentar reduir-ne els efectes és aquest: **les dificultats que pot trobar un infant que pateix a l'hora de comunicar-se i expressar-se** en el seu dia a dia. En el següent punt ens endinsarem més en els objectius de l'aplicació ECO, posant més detall als corresponents al mòdul que he desenvolupat.

## 1.4.- Objectius

A l'hora de desenvolupar un projecte d'aquest tipus és molt important fixar objectius molt concrets, granulars i sobretot mesurables i comprovables. Sovint es peca d'abstracció i generalització a l'hora d'establir-los, de manera que resulta complicat concloure en si s'han assolit o no.

Així doncs, en primer lloc donaré una breu explicació de què és el que pretén l'aplicació en si i després passaré a enumerar els objectius concrets als quals m'he dirigit a assolir amb el meu mòdul i que són, al cap i a la fi, els autèntics objectius del projecte.



Imatge 7: Objectius

### 1.4.1.- Propòsit general de l'aplicació ECO

Com comentàvem abans, el problema al qual ECO vol fer front és el de les dificultats que puguin tenir els nens autistes a l'hora de comunicar-se, de manera que busca ser una eina que serveixi d'ajut en aquest aspecte, presentant a l'usuari tot un seguit de graelles d'elements que poden ser **pictogrames amb so o vídeos classificats en diferents categories**.

La idea és que aquests elements representin clarament i de forma senzilla objectes o accions que formin part dels diferents àmbits o entorns quotidians de la vida d'un nen (casa, escola, menjar, colors, animals), per tal que aquest pugui **veure'ls, entendre'ls i seleccionar-los i així expressar una necessitat** o qualsevol cosa que vulgui

comunicar. A més, es dona la possibilitat de crear elements, ja sigui seleccionant imatges, vídeos i/o sons, o bé capturant-los amb la càmera i la gravadora del dispositiu en qüestió, si en té (evidentment aquesta última funcionalitat és pels pares, tutors, etc).

#### 1.4.2.- Objectius del mòdul propi

Així doncs, la nova idea que s'introdueix amb el meu mòdul és que l'usuari pugui **construir “frases” o conjunts d'elements tot seleccionant i agrupant els diferents pictogrames**, podent expressar o transmetre així missatges més complexos. En aquest cas, a més de poder veure i seleccionar els elements, l'usuari té la opció d'anar-los col·locant a l'espai de la dreta de la graella formant una llista d'elements, i en acabat, poder reproduir-la. Més endavant, en l'apartat de requeriments del producte, veurem en més detall què aporta el mòdul a l'aplicació.

Fruit d'aquesta reorientació, doncs, he dividit els objectius en dos àmbits diferenciats. D'aquesta manera, a banda de l'objectiu principal de **compliment dels diferents requeriments nous** (que veurem més endavant), veurem uns altres sis objectius separats en quant a la relació mòdul – usuari i a la relació mòdul – aplicació:

##### **Objectius mòdul – usuari:**

- 1.- Obtenir una interfície intuïtiva. Que l'usuari compregui en tot moment què està fent i quins efectes causen les seves interaccions.
- 2.- Poblar les interfícies amb els elements justos i imprescindibles. No carregar l'espai amb imatges, textos o detalls innecessaris.
- 3.- Donar suport visual amb animacions gràfiques. Sovint una petita animació pot ajudar a traslladar el significat de l'acció que acaba de tenir lloc en l'aplicació i alhora fa que sigui més amable visualment per a l'usuari. Important no excedir-se en l'elaboració en aquest aspecte tampoc.

##### **Objectius mòdul – aplicació:**

- 1.- Acoblar el mòdul a l'aplicació de forma òptima. Estem parlant d'una aplicació desenvolupada en entorn professional, amb estructures de dades i elements de programació avançada. No és senzill aconseguir acoblar un mòdul aliè així com així, cal tenir clar com aquest interacciona amb la aplicació pare.
- 2.- Respectar la interfície. És a dir, no allunyar-se de com l'aplicació ja definia el disseny, l'estructuració i comportament dels elements en les diferents pantalles. En aquest sentit, volem que de cara als usuaris sigui invisible aquest canvi (quan passin a la pantalla corresponent al mòdul del projecte).
- 3.- Respectar la programació. Estructurar i desenvolupar el mòdul de la manera més similar i acoblable possible a l'aplicació pare, fer ús dels elements i estructures que ja utilitzi l'aplicació. Important per a l'acoblament òptim del mòdul però també de cara a possible treball futur.

## 1.5.- Contingut de la memòria

Per acabar aquest primer punt introductori, abans de continuar, veurem com està estructurada aquesta memòria, de quines parts consta i quins elements són necessaris per a construir un projecte d'aquest tipus.

- Després de la **introducció**, trobem un apartat típic dels projectes d'enginyeria que s'anomena **avaluació tecnològica**. En ell, s'hi expliquen quines opcions s'han valorat, analitzat i escollit en quant a totes les tecnologies implicades, com ara llenguatges de programació, sistemes gestors de bases de dades, sistemes operatius, etc. A més, hi inclouré una breu explicació del programari que he utilitzat per a desenvolupar el projecte.
- **Planificació i desenvolupament** del projecte. Per acabar aquesta primera part més introductòria veurem com he anat dividint la càrrega de feina en diferents paquets de treball i com la he anat disposant al llarg de la durada del projecte. Per fer-ho, em recolzaré en un seguit de diagrames, gràfics i calendaris que ajudaran a fer-ho tot més visual. A més, veurem els diferents agents implicats en el desenvolupament del projecte i la metodologia utilitzada.
- Com a punt de transició entre la primera part introductòria i entrant ja en el que és el projecte en si, el cos de la memòria, començarem veient **l'anàlisi de requeriments**. És a dir, enumerar i descriure totes les característiques desitjades en el mòdul per part del client, que en aquest cas són els professors de la UPC, especialment un. Veurem requeriments de funcionalitat, requeriments d'interfície, etc.
- **Especificació**. En enginyeria informàtica, especificació fa referència al conjunt d'elements que defineixen el problema al qual fem front. Així doncs, tindrem tot un conjunt de diagrames i components per veure tant un model estàtic de la situació (conceptes, relacions) com un de dinàmic (casos d'ús, esdeveniments del sistema).
- **Disseny**. En el disseny es defineix i descriu la solució al problema modelat en l'especificació, és a dir, com interactuen els elements plantejats per a resoldre'l, per al funcionament del mòdul. A més, veurem el disseny de la interfície del mòdul i de les interaccions establertes.
- **Implementació**. A continuació es veuen els mecanismes i mètodes utilitzats per a la implementació del mòdul. És a dir, si en el disseny veiem el què, aquí veiem el com. Tant de cadascun dels elements que el conformen com de les diferents interaccions que es permeten a l'usuari. A més a més, en aquest apartat, definirem diferents conceptes relacionats amb la programació Android que poden anar apareixent al llarg de tot el projecte, especialment en aquest punt.
- Per anar entrant en la part concloent del projecte, començarem repassant les **conclusions** a les que he arribat després del desenvolupament d'aquest projecte juntament amb una valoració de cadascun dels objectius establerts, així com dels requeriments, justificant i analitzant el seu grau de compliment.
- I pel que fa a la part final d'aquesta memòria, veurem algunes idees de possible **treball futur**, la bibliografia usada i, per acabar, els **annexos** on hi trobarem l'índex d'imatges incloses en el projecte.

## 2.- AVALUACIÓ TECNOLÒGICA

En els projectes d'enginyeria informàtica, l'objectiu d'aquest apartat és exposar totes les opcions valorades i analitzades així com les decisions preses pel que fa a tot el conjunt de tecnologies utilitzades en la implementació del projecte. Aquestes engloben típicament els llenguatges de programació, els sistemes gestors de bases de dades (SGBD), els servidors web i els sistemes operatius.



*Imatge 8: Tecnologies*

No obstant, donat el fet ja comentat que el projecte que presento en aquesta memòria és un mòdul per a l'aplicació ja existent **Easy Communicator 'ECO'**, en comptes d'una aplicació en si, el marge de valoració i decisió en quant a tecnologies queda sovint limitat i, fins i tot, reduït a zero, atès que la majoria de tries, en aquest sentit, ja estan preses. Tot i així, veurem que alguna possibilitat d'elecció sí que m'ha quedat.

A continuació, doncs, veurem aspectes sobre el sistema operatiu, els llenguatges de programació i el sistema d'emmagatzematge de dades, donat que de servidor web no n'utilitza cap l'aplicació. A més, faré esment del programari que he utilitzat per a desenvolupar el projecte. Vegem-ho, tant els temes sobre els quals he pogut decidir com els que no, justificant-ho en cada cas.

### 2.1.- Sistemes operatius

Començarem pel sistema operatiu, ja que aquesta va ser una elecció que en un principi sí que vaig poder fer, donat que la vaig fer abans de decidir quina aplicació volia desenvolupar, i no hauria acabat fent el projecte que he fet si no hagués pres aquesta decisió.



*Imatge 9: Sistemes operatius*

Des del principi vaig tenir bastant clar que, si el que volia era crear una aplicació per mòbils, la faria per **Android**, segurament pel fet que l'assignatura cursada en la carrera de desenvolupament d'aplicacions mòbils va ser en un 95% en Android. Tot i així vaig voler investigar què suposava intentar fer-la per a algun altre sistema operatiu.

Això sí, si una cosa tenia 100% clara, era que si no la feia per Android, seria per **iOS** (d'Apple) o bé per **Windows Phone** (de Microsoft), els quals són els tres clars punters en quant a sistemes operatius per mòbils del moment (tot i estant aquest últim bastant per sota d'Android i iOS), i la meua intenció era que aquest projecte fos un punt de partida de cara al futur i una experiència útil. Així doncs, no vaig descartar-los, i vaig indagar una mica en què suposaria desenvolupar una aplicació per a aquests altres dos sistemes operatius.

Vaig descartar iOS ràpidament, tot i ser un sistema operatiu molt vistós i amb molta influència sobretot de màrqueting, bàsicament per les complicacions relatives que suposa desenvolupar-ne *apps*. És necessari treballar des d'un equip amb sistema operatiu d'Apple (MAC) o bé recorrent al recurs de la virtualització. A més, cal pagar una quantitat a l'any per a ser desenvolupador en iOS. Aquest últim aspecte no acabava de ser un gran obstacle per al meu cas però va influir en crear una idea d'exclusivitat i tancament entorn a la idea d'iOS que no em va semblar adequada.

En quant a Windows Phone, vaig veure que tenia certes similituds amb Android, però el fet que fos un sistema operatiu menys rodat i sense gaire disponibilitat de guies, a més que el programari per a desenvolupar requerís també de pagament per obtenir-ne el lot complet, em va tirar enrere ràpidament. A més, he de dir que al llarg de la carrera en el meu cervell s'ha anat formant la idea que tot el que vingui de Microsoft comporta més complicacions que no pas facilitats a l'hora de ser un usuari avançat d'informàtica.

En contrapartida a tots els aspectes negatius vistos, us presento el seguit de característiques que em van ajudar a acabar de decidir-me per **Android**:

- Possibilitat de desenvolupar en Windows, Linux i Mac
- Bona documentació
- Sistema operatiu rodat i treballat
- Aposta forta de **Google**
- Experiència (poca, però una mica)
- Ús de llenguatges amb coneixement avançat
- Aplicacions de codi obert



Imatge 10: Android

Així doncs, vaig decidir-me per Android. I sobre la versió, en el moment de començar a treballar en el meu mòdul ECO, estava programada per a versions 4.X.

## 2.2.- Llenguatges de programació



Imatge 11: Java

En aquest cas sí que hi va haver poca discussió. Al treballar en Android ja estava escollint un llenguatge de programació, però sí que va influir en la decisió que Android utilitzés **Java**. Així que, en aquest sentit, el moment d'escollir llenguatge de programació per desenvolupar el meu mòdul va ser el mateix en que vaig escollir Android, i no després, com sol passar en la majoria de projectes d'informàtica.

Així doncs, el llenguatge de programació en el qual es treballa a l'hora de desenvolupar aplicacions en Android és Java, un llenguatge de propòsit general i orientat a objectes amb una edat que es podria dir relativament prolongada (va aparèixer al 1995 dissenyat per *Sun Microsystems*), i que avui en dia és un dels llenguatges més utilitzats arreu del món per a tot tipus d'aplicacions.

L'última versió és la 8.60 i es distribueix sota llicències GNU GPL (GNU General Public License, llicència amb llibertat d'ús, estudi, compartició i modificació) i és multiplataforma, la qual cosa ajuda a que el desenvolupament d'aplicacions Android també ho sigui.

A més a més de Java, Android també fa ús d'**XML**, un llenguatge de marques desenvolupat per W3C i que serveix per a definir i emmagatzemar recursos, com poden ser layouts (disposició d'interfícies), strings, etc.



Imatge 12: XML



## 2.3.- Sistemes Gestors de Bases de Dades (SGBD)

Pel que fa a Bases de Dades, és a dir, la persistència, el magatzem de tota la informació de què fa ús l'aplicació, si hagués desenvolupat la meua pròpia *app* sí que hi hauria tingut poder de decisió, però, al desenvolupar-ne un mòdul, aquestes decisions ja estaven preses.

L'aplicació ECO treballa amb dos models de persistència i emmagatzematge de dades, de diferents nivells i amb diferents propòsits.

En el primer nivell, el més baix, es fa ús d'un model de dades relacional implementat mitjançant **SQLite**, un sistema gestor de bases de dades programat en C, multi plataforma i de domini públic.



Imatge 13: SQLite

Pel que fa al nivell superior, consta d'un seguit de classes programades en **Java** que emulen una base de dades i actuen d'intermediari entre l'aplicació i el model SQLite. L'objectiu d'aquest intermediari és, sense representar una gran càrrega de memòria, servir de magatzem per a la informació més usada o la més recent, per tal de reduir temps de consulta. Podríem dir que actua com una espècie de *cache*.

## 2.4.- Programari utilitzat

Per acabar aquest apartat d'avaluació tecnològica donarem un cop d'ull a tot el programari que he fet servir per desenvolupar el mòdul. Parlem bàsicament de dos components: **Eclipse**, l'IDE i **Genymotion**, l'emulador, i també veurem els dispositius físics usats per a fer tests més "reals".

### 2.4.1.- L'IDE

Un IDE (Integrated Design Environment) és la millor opció alhora de desenvolupar una aplicació amb una mica d'envergadura. Proporciona un espai de treball adequat per gestionar tot el contingut dels projectes (un IDE funciona amb projectes, cada projecte és una aplicació, un programa), les classes, mòduls, llibreries, etc. Tot. Permet també una compilació, un enllaçat, una execució i depurat molt més còmodes i àgils que si s'hagués de gestionar tot manualment.

Avui dia, a l'hora de desenvolupar un projecte d'Android, podem parlar, sobretot, de dues opcions principals; **Eclipse** i **Android Studio**.



Imatge 14:  
Android Studio

Al llarg de la meua curta experiència com a programador d'aplicacions Android sempre havia fet servir Android Studio, que és un IDE dissenyat i dedicat, òbviament, a aquest propòsit.



Imatge 15: Eclipse

Eclipse, per la seva banda, podríem dir que és un IDE molt més general, que permet desenvolupar molts més tipus de projectes. Des d'aplicacions d'escriptori fins a projectes web, i amb molta varietat de llenguatges de programació; incloent a més de Java, C, C++, Perl, PHP... En aquest cas, per a programar en Android, cal afegir en forma de plug-in un SDK (Software Development Kit) específic d'Android.

Així doncs, tot semblaria indicar que la millor manera de desenvolupar el meu mòdul era Android Studio; el coneixia, és un programa per a aquest propòsit i just s'havia llançat la primera versió robusta (sempre havia treballat amb versions BETA, prou sòlides, però BETA al cap i a la fi). I de fet aquesta va ser la decisió en primera instància, però el fet que l'aplicació ECO hagués estat programada mitjançant Eclipse, va comportar un seguit de complicacions relacionades amb un component de què fa ús l'Android Studio, el *Gradle*, una eina que treballa tot el tema de la construcció, el compilat i l'execució dels projectes.

Juntament amb els professors desenvolupadors d'ECO vam arribar a la conclusió que, pel bé de l'aplicació i del meu projecte, era millor seguir desenvolupant amb Eclipse, que al cap i a la fi, pel meu propòsit, era pràcticament el mateix, i forçar a que treballés amb el *Gradle* era buscar maldecaps innecessaris. Al cap i a la fi un dels objectius és l'acoblament òptim del meu mòdul amb l'aplicació.

#### 2.4.2.- L'emulador

I un cop vist el component més important i principal, veurem l'emulador (menys important però també necessari) amb el qual he anat provant la feina que anava fent.



Com comentava abans, sempre havia desenvolupat aplicacions Android amb l'Android Studio, un IDE que conté també un component anomenat Android Virtual Device Manager, el qual permet crear i gestionar diferents emuladors de diferents dispositius mòbils reals, tot editant-ne i ajustant-ne les característiques. Així doncs, aquesta era l'eina que coneixia.

Eclipse, com dèiem, precisa de la instal·lació d'un SDK específic per desenvolupar en Android, i, a la seva vegada, també permet la instal·lació de l'AVD Manager. La meua idea, doncs, era fer ús d'aquest gestor, la opció per defecte, però, a instàncies dels professors desenvolupadors d'ECO, vaig decidir-me per provar Genymotion.

Genymotion és una *startup* francesa que bàsicament ofereix un component gestor d'emuladors per Android que està adquirint una importància bastant rellevant avui dia. Sobretot, s'està convertint en una opció cada cop més utilitzada degut a les seves millors prestacions que els emuladors bàsics d'Android en quant a la "suavitat" amb que funcionen els emuladors. Cal entendre que sovint, al virtualitzar dispositius relativament potents, és complicat que aquests funcionin amb fluïdesa, de manera que un emulador que mitigui aquest aspecte és realment un objecte de desig.

Un problema que té Genymotion són els problemes de compatibilitat amb versions antigues d'Android, però en aquest ECO (pensada per a versions 4.x) no es veu afectada. Així doncs, vaig decidir seguir amb les "directrius" dels desenvolupadors d'ECO i fer ús de Genymotion com a gestor d'emuladors.

#### 2.4.3.- Dispositius físics

Finalment, pel que fa als dispositius físics sobre els quals he anat realitzant les proves i els tests, he disposat de dues *tablets* de la UPC, ambdues del mateix model, Google Nexus 10.



Aquests models estan fabricats per Samsung però dissenyats amb les directrius de Google. La pantalla és de 10,055 polzades i té una resolució de 2560x1600 píxels.



*Imatge 17: Nexus 10*

### **3.- PLANIFICACIÓ I DESENVOLUPAMENT**

Com a pas previ abans d'entrar de ple en la part central de la memòria, la corresponent al detall del projecte en si, veurem tot el que és la planificació del projecte i els mètodes de desenvolupament i coordinació que he seguit.



*Imatge 18: Planificació*

En projectes d'envergadura, és important, abans de posar-nos a treballar, fer una divisió de la feina en diferents blocs o **paquets de treball**, tot dividint-la en tasques, així com assignar a aquestes tasques una **calendarització** per tal de poder obtenir un seguiment òptim.

Abans de veure tots aquests components, però, m'agradaria explicar el desenvolupament que ha tingut el projecte en un sentit més històric, ja que penso que ha estat un procés un pèl atípic i val la pena detallar-ho.

#### **3.1.- Històric de desenvolupament**

Es podria dir que el projecte comença, de manera no oficial, al febrer de 2014. Acabat el setè quadrimestre de la carrera, tinc clar que no desenvoluparé el projecte en el període suposat del vuitè quadrimestre (febrer – juliol / 2014), principalment un cop presa la decisió de desenvolupar un projecte en forma d'aplicació Android, donat que penso que encara no tinc prou formació en la matèria com per crear una aplicació del nivell desitjat.

Així doncs, dedico 3 o 4 mesos (fins a mitjans de juny) a dur a terme un seguit d'activitats en forma de treballs optatius relacionats amb Android, amb l'objectiu d'ampliar els meus coneixements i arribar a l'estiu amb la sensació, ja sí, de poder començar a treballar en una idea de projecte. Així ho faig, desenvolupant fins a 4 activitats relacionades amb la personalització de *widgets* i la disposició d'imatges en diferents formats, que al final m'han ajudat en el desenvolupament del meu mòdul.

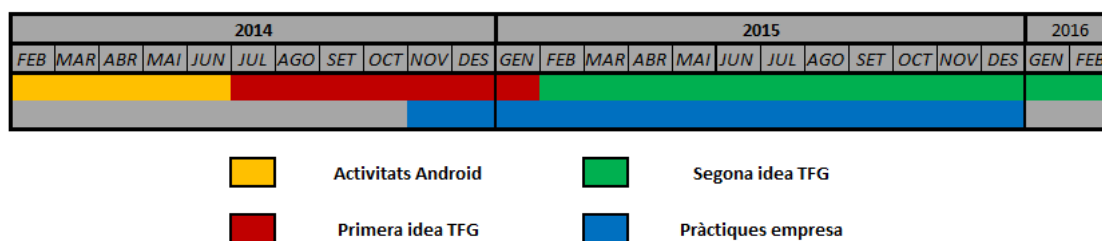
Arribats a juny, començo a pensar en diferents idees que puguin desembocar en un projecte adequat, quan m'arriba la ja esmentada proposta (punt 1.2) per part de la UPC, de desenvolupar una *app* similar a ECO a la vegada que hi afegeixi funcionalitats. Proposta, que com ja he comentat, accepto ja que penso que em pot proporcionar més valor de cara a un futur immediat.

Dedico temps entre juliol i octubre (més en els últims dos mesos) a anar avançant en matèria de projecte així com a investigar una mica l'autisme, amb la idea de presentar el projecte al febrer de 2015, quan a mitjans d'octubre m'arriba una oferta de pràctiques en una empresa. Accepto l'oferta, i prenc la decisió de prorrogar el projecte un quadrimestre més, per tal de poder perllongar l'estada de pràctiques en la empresa i poder seguir amb el projecte d'una manera més òptima.

No obstant, al febrer de 2015 és quan se'm fa la nova proposta de, en comptes de seguir amb la idea d'una aplicació pròpia, col·laborar amb els professors de la UPC desenvolupadors d'ECO, treballant jo en el ja mencionat mòdul acoblable. Des de llavors, i amb l'objectiu d'acabar el projecte al maig de 2015, treballo, doncs, en el meu mòdul, la última i definitiva concepció del projecte.

No obstant, el fet de que les pràctiques m'ocupin bona part del dia i també que la meua dedicació al projecte no és la desitjada ni la necessària, arribats a maig m'adono de que no m'arriben els temps, de manera que em veig obligat a, pel bé del projecte sobretot, allargar-lo un quadrimestre més. Així doncs, dedico també la resta de 2015 (principalment el període setembre-desembre) a desenvolupar el mòdul tenint ja més temps i realitzant una millor i més "estricta" planificació.

Un projecte, que com veieu, ha tingut una durada llarga i peculiar i ha conviscut i evolucionat amb diferents etapes de la meua formació.



Imatge 19: Calendari

### 3.2.- Work Packages

A continuació presento el conjunt de *work packages* (paquets de treball) en què he dividit la càrrega així com la seva funció i les tasques en les que els he descompost. Aquesta divisió és força útil de cara separar la feina i sobretot ajuda a fer una espècie de 'mapa' mental per a veure tot el que cal fer i la seva magnitud.

He fragmentat, doncs, les feines en els següents conjunts de tasques:

**WP1: Gestió administrativa:** Encarregar-se que el projecte es dugui a terme de la millor manera possible.

- T1.1: Planificació: Diferenciació, distribució i calendarització de tasques.
- T1.2: Control de seguiments i terminis: Gestió de fites i informes al llarg del projecte.
- T1.3: Gestió de reunions: Això engloba les diferents trobades que he anat realitzant amb els diferents actors implicats, tant amb el tutor del projecte com amb els professors desenvolupadors d'ECO.

**WP2: Gestió tècnica:** Tasques prèvies o complementàries de suport al projecte.

- T2.1: Documentació sobre l'autisme: Comprensió i assimilació de les característiques principals i aspectes tècnics de l'autisme.
- T2.2: Reporting: Redacció d'informes: Per a ús propi o per a informació de cara al tutor o als desenvolupadors d'ECO.
- T2.3: Recopilació de les fonts d'informació: Cerca i anotació de bibliografia. Documents, articles, guies, vídeos-tutorials, etc. Tot allò consultat per al desenvolupament del projecte.
- T2.4: Confecció de la memòria: Elaboració de les diferents parts de la tesis.

**WP3: Disseny de la interfície:** Definició dels elements que la conformen, la seva disposició i el comportament.

- T3.1: Idea inicial: Esborranys de l'aparença de la pantalla del mòdul.
- T3.2: Reconsideracions i remodelacions: Correccions, canvis i afegits que puguin aparèixer al llarg del projecte.
- T3.3: Disseny funcional: Elaboració del disseny final.

**WP4: Programació del mòdul:** El gruix del projecte, creació del mòdul.

- T4.1: Assimilació continguts: Anàlisi exhaustiu i proves per a la òptima interpretació del codi.
- T4.2: Incorporació i desenvolupament: La programació pròpia, la tasca principal del projecte. Desenvolupament del mòdul i constants proves i anàlisi d'errors.
- T4.3: Documentació: Documentació de tot el codi desenvolupat com a suport per a un anàlisi futur (el corresponent a l'adaptació del mòdul).

**WP5: Validacions:** Tests i anàlisis per a detectar aspectes a corregir i finalment donar el vistiplau al mòdul.

- T5.1: Definició proves: Definició i descripció dels tests a sotmetre.
- T5.2: Realització experiments: Execució de les proves definides.
- T5.3: Resultats: Recopilació i estudi dels resultats obtinguts i valoracions sobre si calen o no modificacions.

### 3.3.- Diagrama de *Gantt*

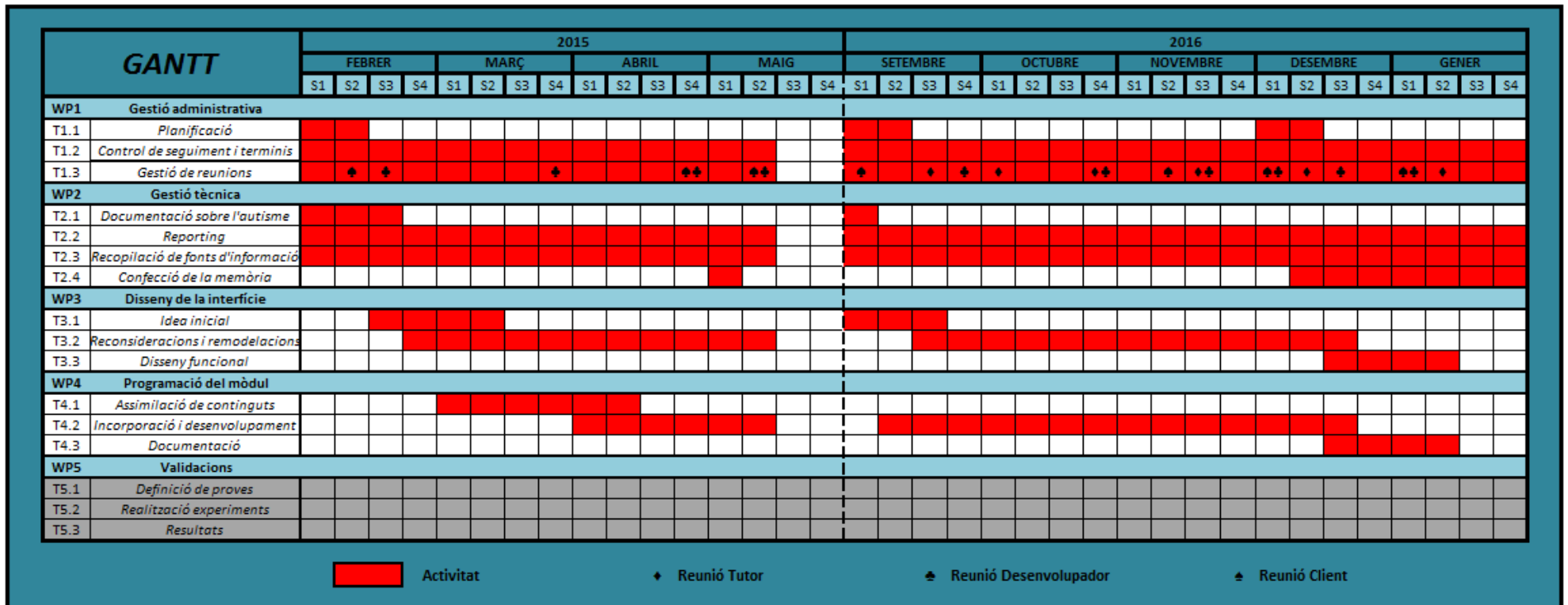
Un cop vist com queda distribuïda la càrrega de feina en paquets de treball, passarem a veure com la he espaiat al llarg de la vida del projecte (etapa febrer/2015 – gener/2016 amb la “pausa” juny-agost, període de poca dedicació).

Per a fer-ho em recolzaré en un tipus de diagrama (següent pàgina, en horitzontal) molt visual que ens ajudarà a veure-ho. A més, hi marcaré les diferents trobades amb els diferents actors implicats (les quals detallaré en l'últim apartat d'aquest punt).

Sobre aquest diagrama, m'agradaria aclarir i/o comentar un parell d'aspectes:

- El primer que crida l'atenció és aquest WP5, el de validació, al qual no se li ha marcat dedicació. Això es deu a que, per motius de temps, no s'ha pogut dur a terme, i queda com a element de treball futur. En aquell apartat (9) ho explicaré amb més detall.
- Destacar també que hi un seguit de tasques que estan disposades de forma contínua, al llarg de tot el projecte. Això es deu a que són tasques de suport que es van duent a terme, no cada dia, però si de forma prou periòdica.
- Com podem veure també, trobem aquest ‘trencament’ temporal que correspon a la segona pròrroga i el nou inici al setembre de 2015, punt en el qual es tornen a realitzar tasques inicials com la documentació sobre diferents aspectes, la planificació, etc.

## DIAGRAMA DE GANTT



Imatge 20: Diagrama Gantt

### 3.4.- Desenvolupament i coordinació

Per anar acabant aquest punt de planificació i desenvolupament, veurem com s'ha anat desenvolupant el projecte en termes de coordinació amb els diferents actors implicats.

En tot treball o projecte final de grau o carrera, evidentment, hi ha la figura del ponent, l'encarregat de fer la tutoria del projecte. A més, però, en aquest projecte cal contemplar dos actors més, els desenvolupadors d'ECO, l'aplicació per la qual he desenvolupat el mòdul.



*Imatge 21: Coordinació*

A continuació descriu, doncs, tots els actors implicats així com detallo la metodologia seguida i el funcionament de les reunions que hem tingut en cada cas.

- Josep Maria Merenciano: **Tutor**, del Departament de Ciències de la Computació (CS). L'encarregat d'anar fent el seguiment del projecte, això sí, amb menys incidència del normal en els aspectes tècnics pel fet que desenvolupés un mòdul d'ECO en comptes d'una aplicació pròpia.

En les diverses reunions que hem tingut ens hem dedicat bàsicament a revisar la feina avançada, la situació respecte a la planificació (a nivell de calendari), i a comentar possibles idees a afegir al mòdul. En la part final del projecte també hem incorporat l'element de la memòria: estructura, continguts, etc.

- Daniel Guasch: **Client**, de la Càtedra d'Accessibilitat (CATAC). És un dels dos creadors d'ECO, però no un desenvolupador pròpiament. El defineixo com a client ja que el seu paper és més de decisió de què incloem en el mòdul i com es comporta.

La finalitat de les reunions ha estat, en aquest cas, similar a les reunions amb un client en qüestions de projectes, validació i peticions. Analitzar la feina feta, valorar la seva viabilitat i parlar de modificacions i correccions a aplicar.

- Israel Martín: **Desenvolupador**, del Departament d'Enginyeria Telemàtica (ENTEL). És l'altre dels dos creadors d'ECO. En aquest cas sí que el defineixo com a desenvolupador ja que és ell pròpiament el desenvolupador d'ECO. El seu paper ha estat iniciar-me en l'aplicació i donar-me suport en qualsevol dubte que tingués en aquest aspecte.

Les reunions amb el desenvolupador, doncs, han estat per si em calia alguna explicació o orientació a nivell purament tècnic. Endinsar-nos i explorar les diferents classes i elements que configuren l'aplicació i analitzar la manera òptima per a que hi pogués introduir el meu mòdul.

Dir també que en el Diagrama de Gantt exposat anteriorment s'hi poden veure aquestes reunions que han anat tenint lloc al llarg del projecte i, també, comentar que al llarg del projecte por haver-hi casos que parli en plural ('hem', en comptes de 'he'). Això es deu a que sovint parlaré d'aspectes relacionats a decisions preses de forma conjunta, i segons el context (programació, desenvolupador; decisions sobre la interfície i la naturalesa del mòdul, client; i temes de planificació i sobre aquesta memòria, tutor) es pot tractar d'un actor o un altre.

### 3.5.- Metodologies de desenvolupament



Imatge 22: Metodologies

I, finalment, trobem aquest subapartat en el que definiré de quina manera treballaré per a incloure el meu mòdul en l'aplicació ECO. Com ja he anat comentant, la meva tasca consistirà en construir una extensió per a un projecte ja en marxa, però penso que és important definir tot el marc d'aquesta situació.

Així doncs, ens trobem en una situació en la que, a partir d'un moment donat, en l'aplicació que ha anat desenvolupant una persona (actor desenvolupador) s'hi ha de posar a treballar una altra persona (jo) per a implementar-hi el meu mòdul, a la vegada que aquesta primera persona hi segueix treballant.

Per a aquestes situacions, existeixen un seguit de plataformes que consisteixen en un repositori distribuït i que proveeixen un entorn col·laboratiu. En aquest entorn, es permet anar fent modificacions sobre un mateix codi per diferents camins, i la mateixa eina s'encarrega de fer aquesta unió (*merge*) de forma automàtica.

No obstant, aquesta eina és útil si les modificacions són relativament petites i els *merge* són bastant constants. Pel que fa a la nostra situació, entenem que per a poder treballar amb més llibertat (cadascú té la seva feina a més a més d'aquest projecte) i minimitzar la coordinació, és millor prescindir d'aquest tipus d'eines i, un cop finalitzat el projecte, fer aquesta unió de forma manual, que a més ajudarà a la comprensió del nou contingut.

Així doncs, la decisió en quant a metodologia a seguir és agafar la versió més actual d'ECO i treballar-hi "a sobre", i més endavant, coordinar-nos per a implementar les meves modificacions en la versió actual d'aquest futur. En l'apartat de treball futur es descriurà amb més detall aquest procés.

## 4.- ANÀLISI DE REQUERIMENTS

En el subapartat d'objectius, en l'apartat d'introducció, hem definit a nivell general què ha d'aportar el nou mòdul a l'app ECO. En els objectius (els objectius de cara a fora, de cara a l'usuari), hem deixat clar que volem una interfície intuïtiva, una distribució òptima dels elements en l'espai (la pantalla) i recolzar-nos en les animacions per a fer-ho més visual.



Imatge 23:  
Requeriments

Ara, però, ja entrant en el detall del projecte, veurem exactament què volem que l'usuari pugui fer a través d'aquest nou mòdul, de quines interaccions ha de poder disposar, quines accions i canvis ha de comportar el fet d'interaccionar amb l'aplicació de les diferents maneres i al cap i a la fi, cadascun dels aspectes que ha d'incloure.

Per a entrar en context, però, primer veurem què pot fer l'usuari amb l'app ECO abans de la incorporació del mòdul.

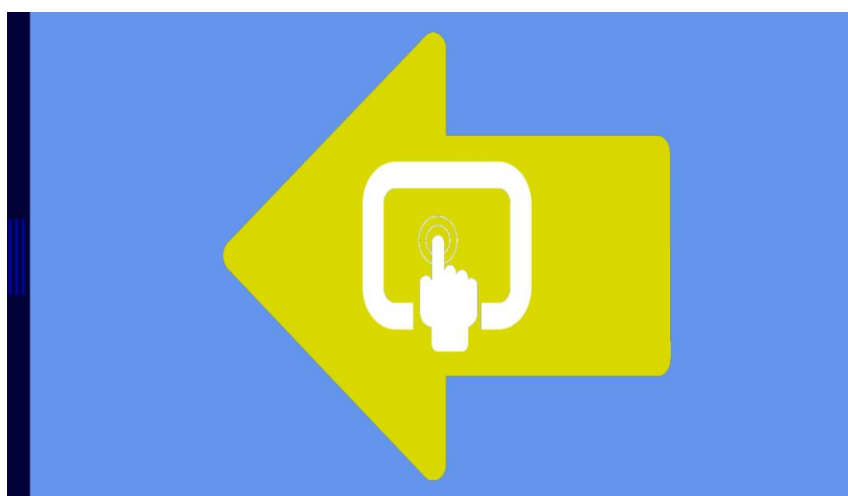
### 4.1.- Escenari inicial



Es podria dir que l'aplicació ECO diferencia dos nivells o tipus d'usuaris; administradors i usuaris. A nivell d'administrador, l'aplicació ofereix diverses opcions: creació d'usuaris, edició de preferències, creació de categories, importar/exportar perfils, etc.

Imatge 24: ECO

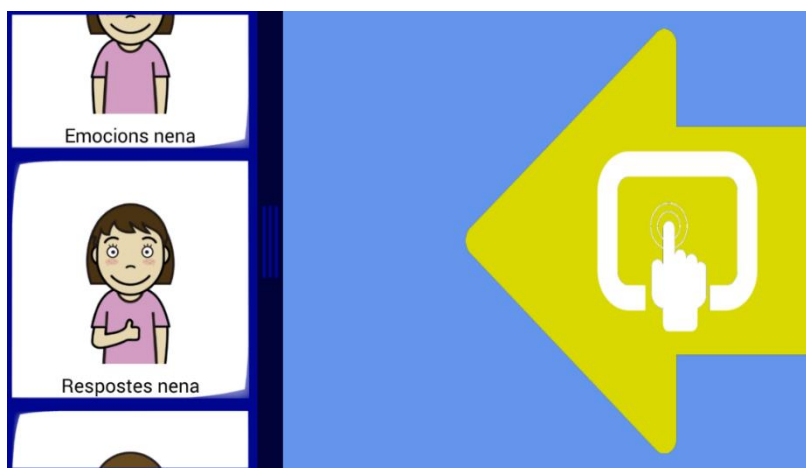
No obstant, l'apartat que a nosaltres ens interessa, ja que al cap i a la fi és el que modificarem, o més ben dit, ampliarem, és el d'**usuari**. En iniciar l'aplicació ens apareix el llistat amb els usuaris existents en format de *view pager*. Al seleccionar-ne un (clic), entrem en la part d'usuari d'aquell usuari, i d'entrada ens trobem amb aquesta pantalla:



Imatge 25: Pantalla inicial

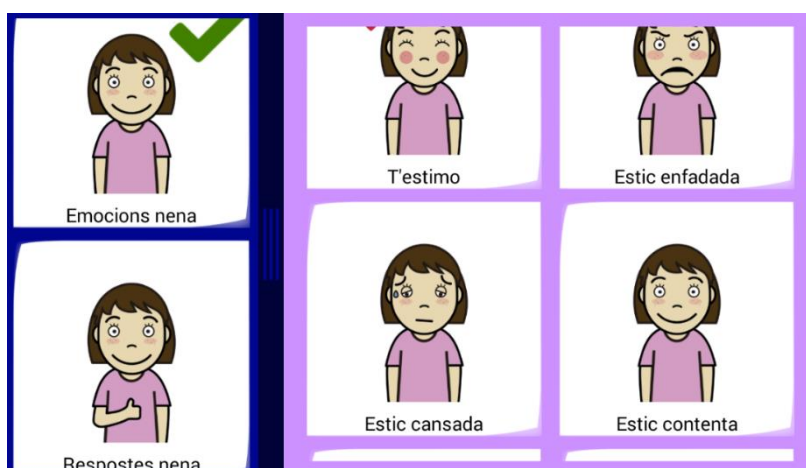
Aquesta pantalla inicial de l'usuari no té massa contingut, simplement és per a seleccionar una de les categories que pugui tenir aquest usuari. Per a veure el llistat, cal (com suggereix el dibuix de fons de la pantalla) arrossegar la pestanya vertical de l'esquerra cap a la dreta per a deixar-lo al descobert:





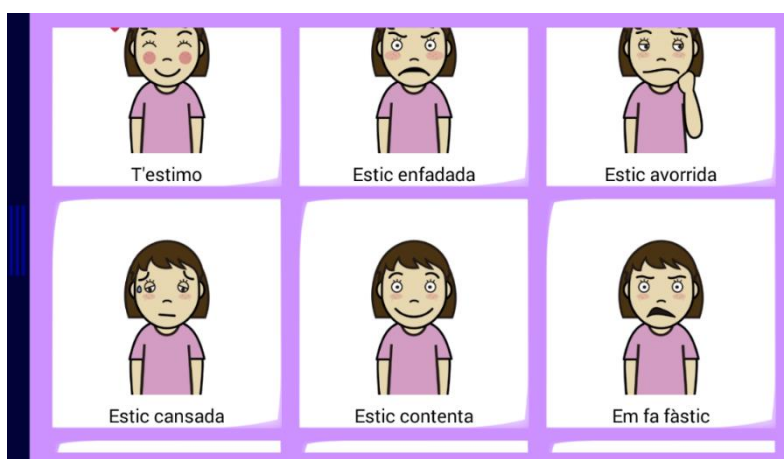
Imatge 26: Llistat categories

Un cop mostrada la llista de categories (l'últim element de la qual és l'enllaç per a tornar enrere) podem seleccionar-ne una, tot fent clic, i veurem que a la part dreta de la pantalla hi apareix una graella d'imatges i la categoria queda marcada en la llista:



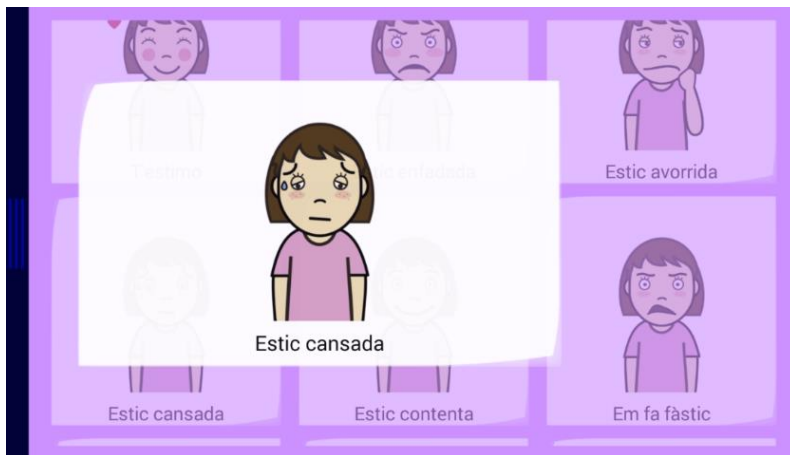
Imatge 27: Categoria seleccionada

En aquesta pantalla tenim una graella conformada pels diferents pictogrames (d'ara en endavant en parlarem com a **missatges**) que conté la categoria que hem seleccionat. Un cop amagada la llista de categories (arrossegant de nou) aquesta graella pot desplaçar-se de manera vertical per anar mostrant els diferents missatges.



Imatge 28: Missatges categoria

A més, tenim la opció de seleccionar un missatge fent clic i contemplar la seva conseqüent reproducció, fent ús d'una nova imatge, que després d'una animació d'ampliació des del lloc del missatge triat i fins a ocupar gairebé la totalitat de la pantalla, executa la reproducció d'aquest i en acabat, mitjançant una animació de reducció fins al lloc original del missatge, desapareix. En les següents imatges podem veure el procés d'ampliació:



*Imatge 29: Ampliació missatge*



*Imatge 30: Reproducció missatge*

Evidentment, des d'aquesta pantalla també podem procedir a canviar de categoria, així com tornar al menú de selecció d'usuari de la mateixa manera que en l'anterior, desplegant el menú de la pestanya de l'esquerra.

## 4.2.- Noves funcionalitats

Un cop situats, ja podem entrar a veure què ha de poder oferir l'apartat d'usuari a partir del mòdul a implementar. Cal dir que, en quant a modificació de pantalles, al cap i a la fi, l'única que veurà canvis en la seva interfície i comportaments és la pantalla de categoria o selecció de missatges, de manera que totes les noves idees van encaminades cap aquesta pantalla.

El primer que cal tenir present és que a partir d'ara introduïrem el concepte de **frase**, o, el que és el mateix, un conjunt de missatges. Sobre aquesta frase es vol que s'hi puguin afegir missatges (a un lloc concret o al final), eliminar-ne un element concret o eliminar-la tota (buidar-la) i reproduir-la de manera seqüencial (un per un els elements que la conformen). A més, com comentàvem abans, és necessari acompanyar tot aquest seguit de funcionalitats noves amb animacions que ajudin a l'usuari (recordo que el nostre usuari n'és un de ben especial) a comprendre tot el que va succeint.

Per tal de veure tots aquests nous requeriments pactats amb l'actor *Client* de la UPC, els disposarem en una llista:

- **Col·locació i format:** Així doncs, volem representar una frase. Donant un cop d'ull a la pantalla de la categoria i tenint en compte que la majoria dels dispositius per als quals està pensada l'aplicació són més amples que alts, ràpidament ens ve a la ment que la col·locació òptima seria a un costat, i en forma vertical.

És cert que es pot distanciar una mica del sentit lògic d'una frase (l'escriptura és horitzontal) però prioritzem la disposició visual de la interfície (queda millor sobretot la graella) i d'aquesta manera acompanyem també a la graella, ja que el desplaçament (quan hi hagi prou elements a la frase com per que ni pugui haver) serà vertical també.

Pel que fa al costat, tenint en compte que el menú de categories es desplega des de l'esquerra, pensem que el millor és col·locar-lo al costat dret.

- **Afegiments i insercions:** Per a l'afegiment, en un primer moment pensem de fer-ho de manera que en fer clic en un missatge aquest es reproduïx, com abans, però en acabat que s'afegeixi a la llista, fent que l'animació de reducció sigui cap a aquesta. Després, però, veiem que així ja no hi ha opció de simplement reproduir un missatge, per la qual cosa decidim que per afegir es faci arrossegant un missatge a la llista (frase).

Aquí cal remarcar que la nostra llista segueix un funcionament FIFO (*First In First Out*) de forma que els missatges afegits primer se situaran a dalt i els posteriors s'aniran col·locant a sota.

No obstant, més endavant afegim una nova funcionalitat per la qual s'ha de poder afegir o bé al final o bé en una posició concreta, és a dir, inserir. Aprofitant que el mètode d'afegiment serà el d'arrossegament, definim que allà on es deixi anar el missatge (de la zona de la frase, de la llista) és on es col·locarà.

- **Eliminacions:** Per a eliminar pensem ràpidament també en l'arrossegament, en aquest cas a una imatge d'una paperera que es sobreposés a la graella en el moment de seleccionar un element de la frase. Això per a un element en concret, però també volem un mecanisme per a esborrar la frase sencera.

Pensem en un botó, però ho acabem descartant pel fet que sigui una interacció massa simple (es podria pressionar el botó per accident) i per a una acció tant "grossa" com és esborrar-ho tot.

Finalment, trobem una altra manera millor que és forçar que la frase ja tingui un primer element per defecte que sigui com una espècie de capçalera d'aquesta (ja que està situat a dalt). La qüestió es que en arrossegar aquest element arrosseguis tota la frase, que visualment quedi clar que deixant-ho a la paperera l'esborrés sencera.

Un altre motiu de pes per a fer-ho així és que segueixes la metodologia de l'eliminació individual i que el fet que aquest element formi part de la frase, en comptes de ser un botó que ocupa un espai fix, fa que es pugui amagar (fent *scroll* a la llista de la frase) per a poder veure “més frase”.

- **Reproducció:** La reproducció en un principi la ideem com una successió de les reproduccions individuals dels elements que componen la frase, això sí, amb una sola animació d'augment al principi i la de reducció en acabat.

No obstant, volem diferenciar la reproducció individual de la grupal a nivell visual, de manera que fem que en comptes d'ocupar tota la pantalla (o gairebé), en el cas de la reproducció de la frase no tapi la llista, només la graella. Això ajuda també a veure per quin element de la frase vas, ja que pensem també en alguna manera d'anar marcant la llista a mesura que es vagin succeint les reproduccions.

Pel que fa a la manera d'accionar aquesta reproducció grupal, pensem en un botó de reproduir (la típica imatge de *play*), i que quan s'estigui reproduint s'hi sobreposi un altre botó per poder sortir de la reproducció en qualsevol moment i tornar a la pantalla original (de la categoria).

- **Animacions:** Vistes les funcionalitats noves, cal parlar també de d'animacions, per tal de, com ja hem esmentat, acompanyar en tot moment els processos de manera visual. A més a més de les d'augment i reducció en les reproduccions, n'incorporem tres més:
  - Afegiment / inserció: Per tal que els elements o missatges de la frase no apareguin de cop en aquesta, introduïm una animació suau d'aparició progressiva (1 segon) en el lloc corresponent. De manera que al principi només es veu un espai buit i a poc a poc hi va apareixent el missatge
  - Eliminació: El cas invers de l'afegiment i la inserció, en aquest cas fem que un missatge no desaparegui simplement, sinó que ho faci progressivament.
  - Transició: Entre elements en la reproducció de la frase, alguna cosa senzilla (com una pampalluga) que faci notar que s'ha canviat de missatge.
- **Indicadors de la frase:** A més a més, pensem que estaria bé “informar” a l'usuari de si hi ha més elements en la frase tant per sobre com per sota del que ell veu en un moment donat (insistim en la importància de pensar en el tipus d'usuari). Col·loquem dues fletxes a la part superior i inferior dreta per indicar-ho, respectivament. Òbviament, aquestes fletxetes només són visibles quan pertoca, que és quan hi ha elements per sobre o per sota amagats.

## **5.- ESPECIFICACIÓ**

Tal i com hem vist en l'apartat anterior, l'anàlisi de requeriments, la "zona" d'ECO a la qual hem d'aportar noves funcionalitats és la d'usuari. D'aquesta manera, en aquest apartat ens centrarem en la situació que es planteja, a nivell d'especificació, en aquest bloc de l'aplicació, en comptes de fer-ho en la totalitat del sistema.

Per a veure-ho en detall i comprendre millor com és la nostra incidència, una mica en la línia del desenvolupament del punt anterior, oferirem dues perspectives: la situació o escenari inicial i la referent, ja sí, a la incorporació del nostre mòdul i les seves ampliacions.

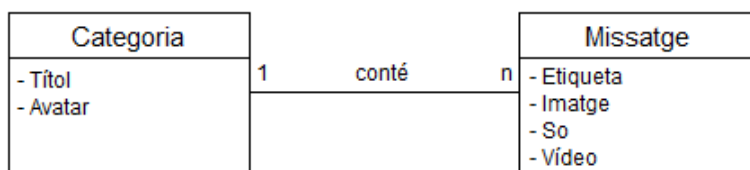
En cada cas, ens situarem en la definició del problema tant a nivell estàtic, veient el model conceptual (diagrama conceptual, glossari i restriccions semàntiques), com a nivell dinàmic, definint i descrivint els casos d'ús que té ECO en cadascuna de les dues perspectives.

### **5.1.- Escenari inicial**

Comencem, doncs, per aquest escenari inicial que, pel que fa a l'especificació del nostre problema, tan sols ens hem de centrar en dos conceptes i que, com veurem més endavant, al cap i a la fi només incidirem sobre un d'ells.

#### **5.1.1.- Diagrama conceptual**

En aquest senzill diagrama conceptual podem veure-hi els conceptes 'Categoria' i 'Missatge', juntament amb els seus atributs i les relacions corresponents:



*Imatge 31: Diagrama Conceptual*

Pot sobtar una mica el fet que aquest diagrama sigui tan reduït, però insistim que el que pretenem modelar són simplement els elements que necessitem definir de cara a la nostra incorporació.

#### **5.1.2.- Glossari**

##### **Conceptes:**

- **Missatge:** La nostra peça bàsica. Un missatge representa una acció o un element concret o abstracte que, mitjançant els seus atributs, pretén transmetre alguna idea o concepte recognoscible per a l'usuari.

- *Categoria*: Agrupació de missatges segons la seva temàtica o context.

#### **Atributs:**

- *Missatge.Etiqueta*: El text que apareix en la part inferior d'un missatge, el seu nom.
- *Missatge.Imatge*: La imatge/avatar del missatge en sí.
- *Missatge.So*: El so que es reproduïx en reproduir un missatge (pot ser que no n'hi hagi).
- *Missatge.Vídeo*: El vídeo que es reproduïx en reproduir un missatge (pot ser que no n'hi hagi).
- *Categoria.Títol*: El nom de la categoria, que identifica la classe dels missatges en concret.
- *Categoria.Avatar*: La imatge que es mostra en el llistat de selecció de categoria.

#### **Relacions:**

- *'conté'*: Fa referència a la relació d'una categoria amb els diferents missatges que ajunta.

### 5.1.3.- Restriccions i suposicions semàntiques

En aquest apartat s'hi acostumen a incloure indicacions o directrius que no es poden extreure directament del diagrama conceptual i el glossari o bé resolen possibles ambigüitats que aquests puguin ocasionar. No obstant, en aquest primer i reduït escenari no en tenim cap. Més endavant, en l'escenari "final", en trobarem alguna.

I fins aquí hem vist el referent al model estàtic. A continuació veurem els casos d'ús, els quals formen el model dinàmic i representen les accions o propòsits que l'usuari pot voler realitzar en el sistema. Recordem de nou que veurem els relatius tan sols a la secció d'usuari d'ECO.

### 5.1.4.- Casos d'Ús

Pel que fa a la nostra zona d'acció, de la qual hem definit el model conceptual, tan sols tenim un cas d'ús, que és la reproducció d'un missatge en concret. A continuació en veiem l'especificació.

#### **Reproduir missatge:**

- Responsabilitat: Dur a terme la reproducció d'un missatge concret seleccionat.
- Seqüenciació d'esdeveniments del sistema: *reproduirMissatge(missatge)*
- **reproduirMissatge(missatge)**
- Arguments: *missatge* (l'identificador del missatge a reproduir)

**PRE:** Existeix un *m:Missatge* corresponent a *missatge*.

No hi ha cap reproducció executant-se.

**POST:** S'ha reproduït el Missatge *m*.

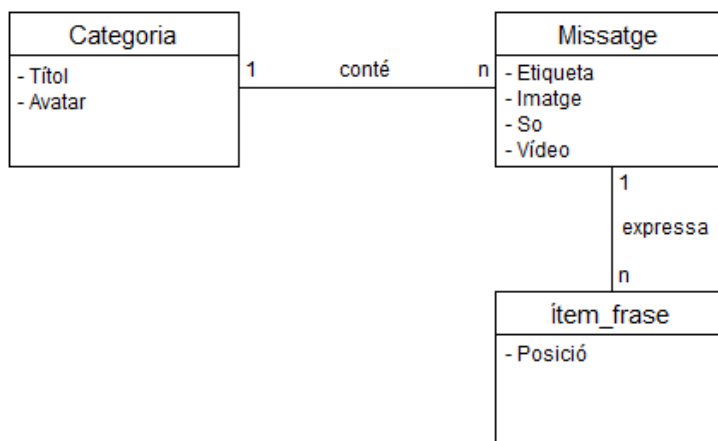
## 5.2.- Nova especificació

A partir d'ara cal tenir en compte que tenim un nou element a modelar el qual incloem amb el nou mòdul, que és la frase. A continuació tornarem a parlar del model conceptual i dels casos d'ús, tenint en compte ara les noves aportacions en aquest nou escenari "final".

### 5.2.1.- Nou diagrama conceptual

Com podem veure en el següent diagrama, el nou concepte per a modelar la frase no és la mateixa frase, sinó aquest nou "*Ítem\_frase*". En un primer moment vaig pensar en modelar aquest concepte (frase), però em vaig adonar que no hi ha un concepte frase en sí, donat que de frase només n'hi ha una en tot moment i, al final, no és més que un conjunt de missatges, que com veurem, tampoc són ben bé objectes del concepte Missatge.

En tot cas, es podria arribar a modelar la frase com un concepte *singleton*, un concepte d'una sola realització, però no és necessari.



Imatge 32: Nou Diagrama Conceptual

Com hem comentat abans, no són ben bé objectes del concepte Missatge el que reuneix una frase, ja que tenim un nou atribut a modelar que és la posició en la frase que ocupa cadascun d'aquests *Ítem\_frase*. A més a més, necessitem també aquesta "realització" de Missatge en la frase per a poder afegir-hi dos ítems d'un mateix Missatge, opció que en cap cas es nega als nostres usuaris.

### 5.2.2.- Ampliació del glossari

#### Conceptes:

- *Ítem\_frase*: Expressió d'un Missatge juntament amb la seva posició.

#### Atributs:

- *Ítem\_frase.Posició*: Posició en la frase en la qual s'ubica aquest ítem.

### Relacions:

- 'expressa': Fa referència a la relació per la qual un Missatge es pot realitzar en diferents Ítem\_frase.

### 5.2.3.- Restriccions i suposicions semàntiques

Ara, en aquest nou escenari, sí que tenim algunes restriccions i suposicions semàntiques. Són les següents:

- No pot haver-hi dos Ítem\_frase amb el mateix valor en l'atribut Posició.
- Tots els valors de Posició dels diferents Ítem\_frase que tenim en un moment donat han de ser valors compresos entre 1 i n, sent n el nombre d'Ítem\_frase que tenim.

### 5.2.4.- Casos d'Ús

Pel que fa als casos d'ús, en aquest nou escenari en trobem un de nou a banda del ja descrit de reproduir un missatge. Ara, tenim també el de crear i reproduir una frase:

#### Crear i reproduir frase:

- Responsabilitat: Dur a terme la creació, construcció i reproducció de la frase.
- Seqüenciació d'esdeveniments del sistema:  
*crearFrase()* + *afegirMissatge(missatge, pos)\** + *reproduirFrase()*

- **crearFrase()**

- Arguments: -

**PRE:** No hi ha cap reproducció executant-se.

**POST:** No tenim cap Ítem\_frase.

- **afegirMissatge(missatge, pos)**

- Arguments: *missatge* (l'identificador del missatge a incloure), *pos* (la posició a col·locar el missatge. Si és *null*, s'afegeix al final)

**PRE:** Existeix un *m:Missatge* corresponent a *missatge*.

No hi ha cap reproducció executant-se.

Tenim Ítem\_frase com a mínim fins a la posició *pos-1*.

**POST:** Tenim un nou *i:Ítem\_frase* corresponent al *Missatge m*.

El valor de l'atribut Posició de l'Ítem\_frase *i* és *pos* i els Ítem\_frase que tenien un valor igual o superior a *pos* (si n'hi havia) han vist el seu atribut Posició incrementat en 1. Si el valor de *pos* és *null*, el valor de l'atribut Posició de l'Ítem\_frase *i* és *n*, sent *n* el nombre d'Ítem\_frase que tenim ara.



- **reproduirFrase()**

- Arguments: -

**PRE:** Tenim, com a mínim, un *i:Ítem\_frase*.

No hi ha cap reproducció executant-se.

**POST:** S'han reproduït tots els *Ítem\_frase* en l'ordre correcte.

Seguim tenint els mateixos *Ítem\_frase*.

Per acabar, cal destacar que la seqüenciació d'aquest cas d'ús definida abans (*crearFrase()* + *afegirMissatge(missatge, pos)\** + *reproduirFrase()*) és una seqüenciació paradigmàtica, és a dir, la seqüenciació que se suposa que ha de dur a terme l'usuari.

No obstant, si volem definir la seqüenciació que abasta totes les possibilitats i opcions, seria aquesta:

$$crearFrase() + (afegirMissatge(missatge, pos) + [afegirMissatge(missatge, pos) | reproduirFrase()])^* + crearFrase()$$

És a dir, un cop tenim la frase creada, l'únic que podem fer és afegir-hi un missatge, però després podem afegir-n'hi més o reproduir-la (es reproduceix sempre i quan no estigui buida) tantes vegades com vulguem. I, en acabat, com veurem més endavant, podem eliminar-la.

Comentar també que pot sobtar una mica el fet que només tinguem un nou cas d'ús. No obstant, així és; d'accions pròpies, com veurem en el següent apartat (disseny), en tenim més, però en els casos d'ús tan sols s'hi inclouen les accions que corresponen als usos per als quals ha estat dissenyat el sistema; en aquest cas, el nostre mòdul.

Així doncs, en el disseny veurem com, a partir d'aquesta definició del problema al qual ens volem enfrontar que hem fet, construïm la solució. A més a més, veurem tot el disseny de la interfície de la nova pantalla de selecció de missatges i totes les interaccions que hi tenim, ja plantejades en l'anàlisi de requeriments.

## 6.- DISSENY

Després d'haver vist l'especificació trobem el disseny. Com ja hem comentat, aquest punt està bastant relacionat amb l'anterior (especificació), ja que a partir dels models definits proposa una solució. Així doncs, si en l'especificació hi trobàvem el model conceptual com a model estàtic i els casos d'ús com a model dinàmic, aquí hi trobem el model de components i el de comportament, respectivament.

A diferència del punt anterior, però, començarem pel model dinàmic; el model de comportament. Això és deu a que a partir del disseny dels casos d'ús obtenim les diferents visibilitats, les quals modelitzen en el disseny el que en l'especificació eren les associacions dels diferents conceptes, ara components. D'aquesta manera, després, trobarem el model de components, el model estàtic del disseny.

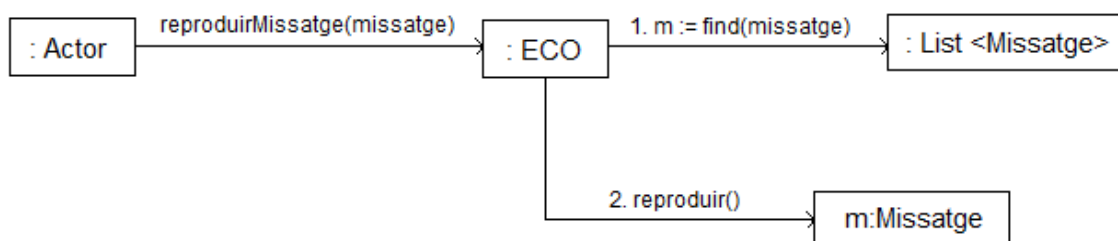
Un cop vist el disseny de la nostra solució, ja entrarem a veure, analitzar i justificar el disseny de la interfície del nostre mòdul així com el de les diferents interaccions que la componen, un dels plats forts d'aquest projecte.

### 6.1.- Diagrames de col·laboració

Així doncs, el en el model de comportament hi trobem el disseny dels casos d'ús, el que es coneixen com a diagrames de col·laboració. En cadascun d'ells, veurem com els diferents components del nostre sistema s'intercanvien missatges per a dur a terme les accions pertinents en cadascun dels esdeveniments de sistema dels nostres casos d'ús.

#### 6.1.1.- Cas d'ús 1: Reproduir missatge

- reproduirMissatge(missatge)



Imatge 33: Reproduir missatge

Aquest primer cas d'ús, que ja teníem en l'escenari inicial, tan sols té un esdeveniment de sistema, el diagrama de col·laboració del qual veiem aquí sobre.

En aquest primer diagrama de col·laboració, doncs, podem observar com la petició introduïda per l'actor arriba al sistema (ECO), el qual s'encarrega de buscar el missatge corresponent al de l'identificador que se li ha fet arribar. Un cop amb el missatge identificat, li envia un missatge per a que es dugui a terme la reproducció d'aquest.

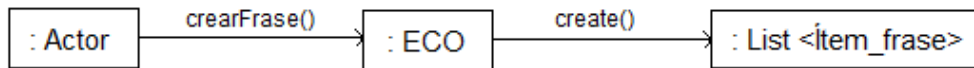
### 6.1.2.- Cas d'ús 2: Crear i reproduir frase

Aquest segon cas d'ús, com hem vist en l'especificació, es compon de la seqüència d'esdeveniments del sistema:

*crearFrase() + afegirMissatge(missatge, pos)\* + reproduirFrase()*

Així doncs, veurem un diagrama de col·laboració per cadascun d'ells:

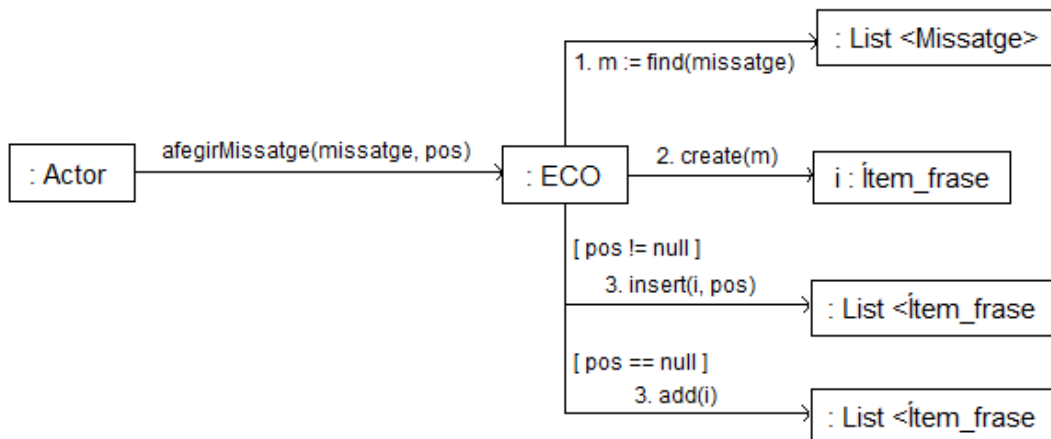
- **crearFrase()**



Imatge 34: Crear frase

Tal i com hem dit en l'especificació, a nivell conceptual la frase no és més que un conjunt de Ítem\_frase, de manera que, en aquest segon diagrama de col·laboració, el sistema se'n crea un multiobjecte en forma de llista, evidentment buit.

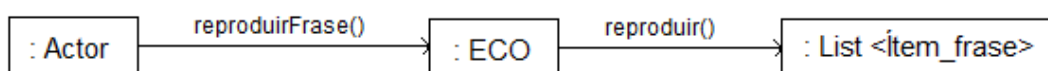
- **afegirMissatge(missatge, pos)**



Imatge 35: Afegir missatge

Aquest tercer diagrama de col·laboració representa la inserció d'un missatge en la nostra frase. Igual que en els anteriors, el missatge arriba al sistema, que en aquest cas s'encarrega de dur a terme tres tasques. En primer lloc, trobar el Missatge corresponent a l'identificador obtingut a través de l'argument de la funció afegirMissatge. Després, s'encarrega de crear un Ítem\_frase corresponent a aquest Missatge i finalment, segons si l'argument pos era diferent o igual a null, el col·loca en aquella posició al multiobjecte de Ítem\_frase o bé el l'afegeix al final.

- **reproduirFrase()**

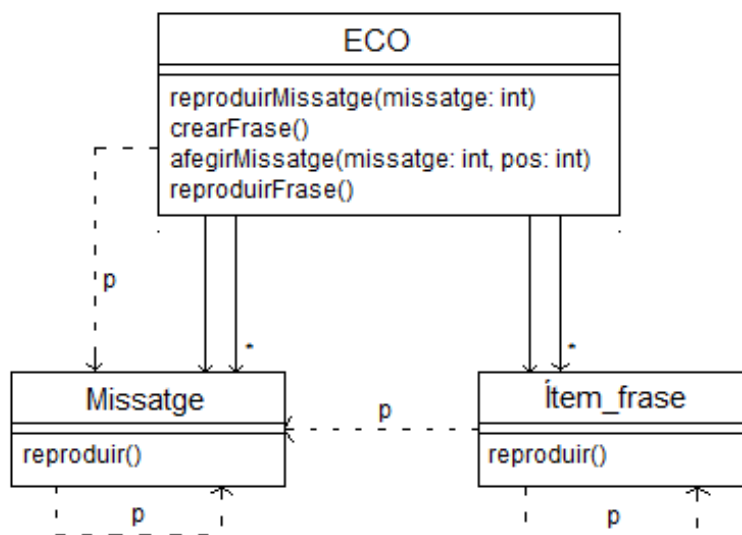


Imatge 36: Reproduir frase

Finalment, tenim aquest quart diagrama, el qual s'assembla molt al primer, degut al fet que representin una tasca similar. En aquest cas, en comptes d'haver de cercar el Missatge a reproduir, el sistema directament envia el missatge al multiobjecte de Ítem\_frase per a que es reproduxeixi tota la frase.

## 6.2.- Model de components

A partir de tots aquests diagrames de col·laboració i del diagrama conceptual, construïm doncs el model de components, amb totes les entitats representades, els seus mètodes i les visibilitats corresponents.



Imatge 37: Model de Components

D'aquest diagrama destaquem el fet que aparegui aquest component ECO, el qual representa el sistema al que li arriben les diferents peticions i missatges, i per altra banda destaquem també el fet que no aparegui el component 'Categoria', que sí que era al diagrama conceptual. Això es deu a què, com ja dèiem en l'especificació, el concepte important era Missatge, i aquí veiem que és el que té interaccions. Categoria, per la seva banda, no tenia cap interacció amb cap component, de manera que hem optat per no afegir-lo.

## 6.3.- Disseny de la interfície

Deixem enrere els temes més tècnics del disseny i entrem en la part més visual que és la del disseny de la nova interfície d'aquesta pantalla de l'usuari, la de categoria o selecció de missatge. Primer de tot ens centrarem en la pròpia naturalesa i la ubicació en el mòdul dels diferents elements que constitueixen aquesta nova interfície i després passarem a comentar ja les interaccions que hi pot dur a terme l'usuari.

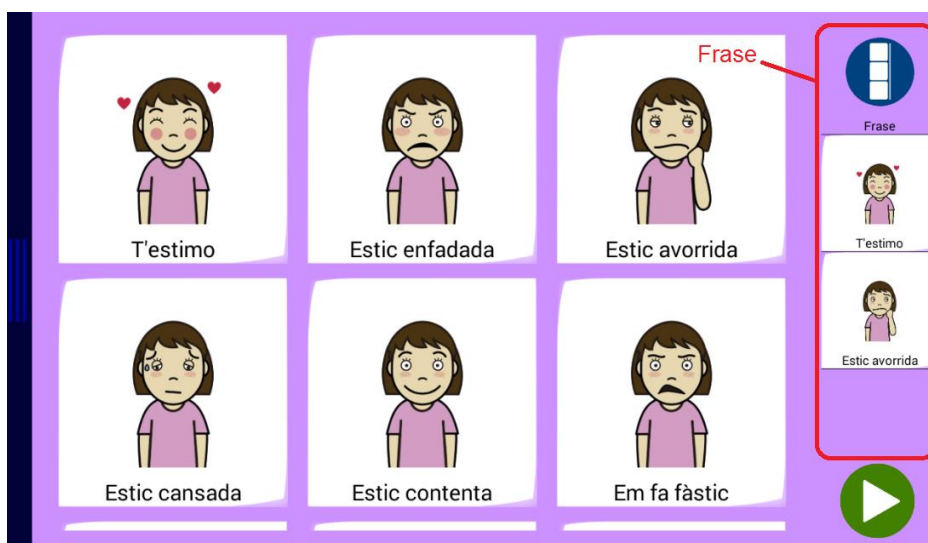
### 6.3.1.- Disseny dels components

Evidentment, el primer component de què hem de parlar, el més important dels que introduïm en la interfície, és la **frase** o el conjunt d'Ítem\_frase.

Primer de tot, m'agradaria comentar que el fet de deixar enrere la especificació i el disseny tècnics suposa entrar en una nova "dimensió" o marc en la qual sí que podem començar a parlar de la frase com a conjunt de missatges, ja que parlarem de la interfície i, a nivell visual, els elements de què es compona la frase són els mateixos que omplen la graella i agrupen cada categoria.

Per tant, d'ara en endavant, parlarem dels elements de la frase com a missatges, tot i havent deixat clar que no són realitzacions del concepte *Missatge* definit en l'especificació, sinó que ho són del concepte *Ítem\_frase*.

Parlant de la frase, doncs, en l'apartat d'anàlisi de requeriments ja hem comentat que el format i la col·locació òptima d'aquest element haurien de ser en vertical i a la part dreta de la pantalla. De manera que ens trobem amb aquesta disposició:



Imatge 38: Frase

Evidentment, el fet d'introduir aquest nou element fa que perdem espai pel que fa a la graella de missatges. No obstant, aquest és un fet inevitable, ja que veiem del tot necessari que l'usuari pugui veure en tot moment tant la graella com la frase. Recordem una vegada més que el nostre usuari té unes característiques especials, i fer que el nou element principal pugui estar amagat o parcialment amagat no seria una bona idea.

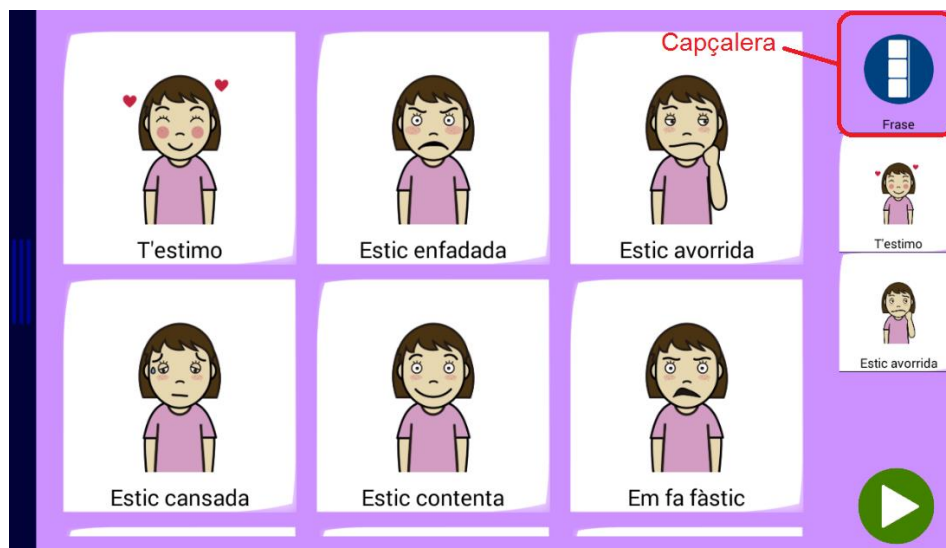
Pel que fa a l'orientació, reiterem que hem decidit optar per aquesta col·locació vertical per dos motius. En primer lloc, l'ocupació d'espai respecte a la graella és més raonable (si fos en horitzontal, tot i que es podrien veure més elements de la frase, ens menjaríem més espai de la graella), i en segon lloc, d'aquesta manera fem que l'usuari tingui el mateix tipus de desplaçament (scroll); vertical, tant per la graella com per la frase.

Com ja hem dit també en l'anàlisi de requeriments, la col·locació a la dreta es deu a que, d'aquesta manera, "allunyem" les dues llistes que hi podem trobar en aquesta pantalla (el menú de selecció de categories i la frase), ja que mai es veuran les dues alhora. Pel que fa al funcionament de la frase, dir que evidentment seguirà un patró

FIFO (First In, First Out), és a dir, que afegint (no inserint) un missatge, aquest es col·loca al final.

Cal destacar també, en aquest context, un element de la frase que ens pot haver sorprès. Es tracta d'aquest element de **capçalera** que sembla un missatge més però no ho és. Aquest element, com ja veurem més endavant, té la utilitat de poder esborrar la frase, sempre i quan contingui algun missatge.

De la mateixa manera que interactuant amb un sol element amb la finalitat d'esborrar-lo, fer-ho amb aquest element que "representa" la frase en la seva totalitat implicarà esborrar la frase sencera. Quan parlem d'eliminacions i esborrat, però, ja en veurem més detalls.



Imatge 39: Capçalera frase

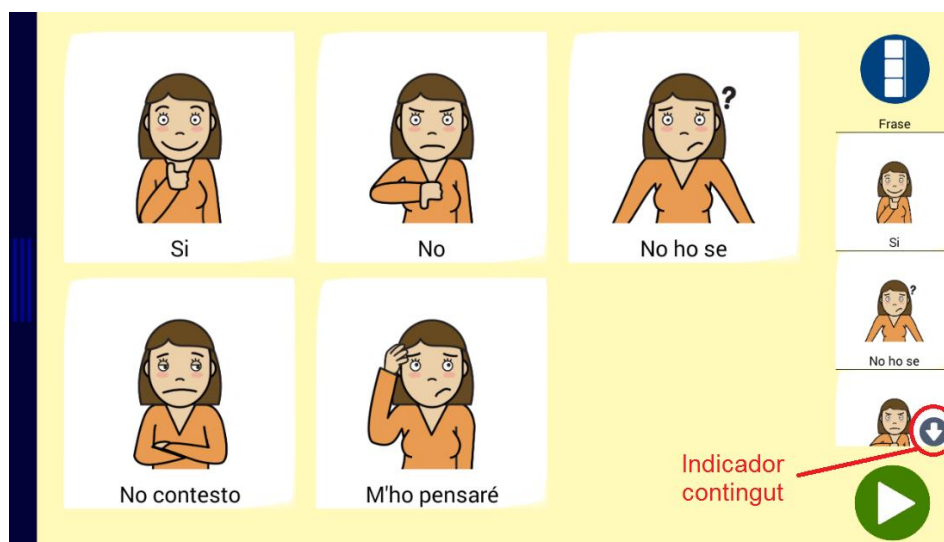
És important destacar el fet que col·locar-lo com a un element més de la frase i no independentment es deu a la possibilitat "d'amagar-lo", fent scroll a la frase per així poder veure'n més missatges reals.

Per acabar amb la frase en si, veurem dos elements més que fins ara havien quedat amagats, o més ben dit, invisibles. En un primer moment, vam pensar que les frases que construïren els usuaris serien de tres, quatre o fins a cinc elements (concepció lògica d'una frase: subjecte, acció, objecte i complements). No obstant, evidentment res impedeix fer-ne de més llargues, ja que no hi ha límit de missatges en una frase.

Per això, i donat el tipus d'usuari al qual va dirigida l'aplicació, vam pensar que la possibilitat de "perdre's" en la frase quan aquesta tingués un nombre suficientment elevat de missatges era un punt a resoldre molt important.

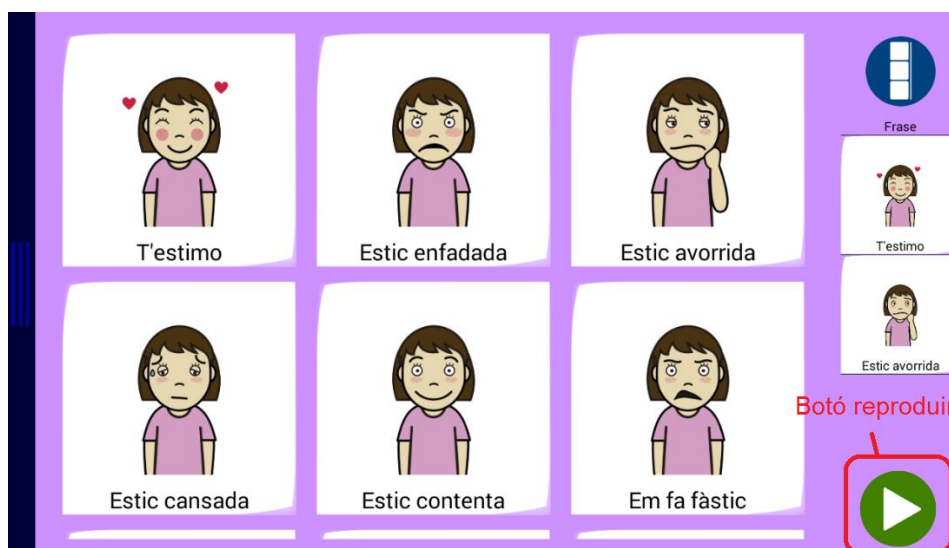
És per aquest motiu que vam decidir afegir dos **indicadors** en forma de fletxes que indiquin, sobre el **contingut** de la frase que es veu en un moment donat, si hi ha més elements tant per sobre com per sota.

Pensem que aquests indicadors poden arribar a ser útils si, insistim, un usuari construeix frases "llargues" o simplement li costés més del normal recordar els elements de la frase i ubicar-se. Evidentment no hem volgut que aquests indicadors destaquin massa, però sí prou com perquè l'usuari se n'adoni que hi són.



Imatge 40: Indicadors contingut

I, per últim, trobem aquest últim element situat a sota de la frase que no és altre que el **botó** per a accionar la **reproducció** de la frase (si en tenim, de frase).



Imatge 41: Botó reproduir

La decisió de col·locar-lo a sota neix en un primer moment en què teníem pensats dos botons; un de reproducció i un altre d'eliminació. Un el vam col·locar a sota (reproducció) i l'altre a sobre (eliminació) de la frase. La idea era tenir-los separats perquè l'usuari no es confongués a l'hora de seleccionar l'acció a dur a terme.

Més endavant, però, vam prendre la decisió de prescindir d'aquest segon botó d'eliminació i vam substituir-lo per la capçalera de la frase que ja hem comentat (per un motiu de guany d'espai, bàsicament) i les seves interaccions corresponents per a dur a terme aquesta acció d'esborrat.

En aquell moment, la decisió de separar ambdós elements (capçalera i botó de reproducció) es va mantenir pel mateix motiu de separar espacialment les dues accions, quedant doncs el botó de reproducció situat a baix.



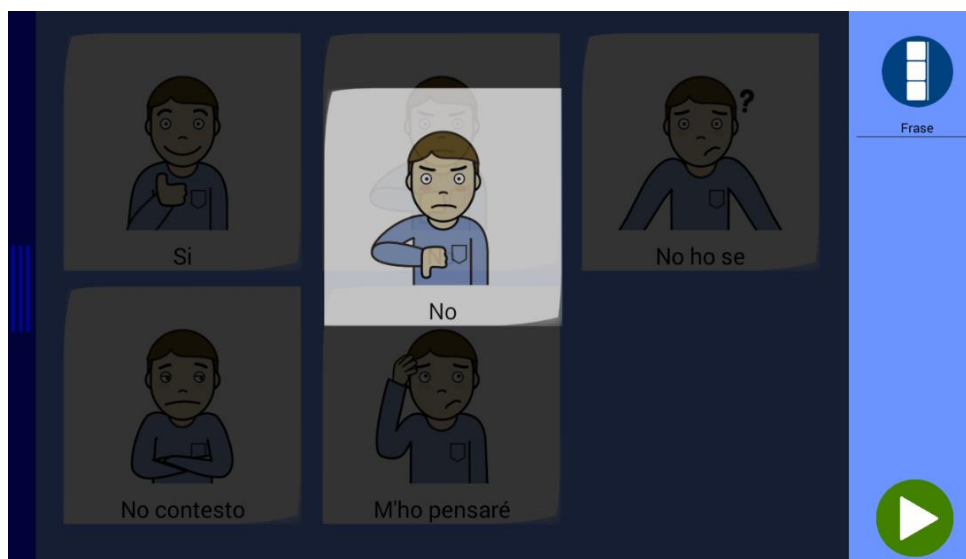
Com hem pogut veure, tots els nous elements inclosos els hem col·locat aprofitant l'espai en un primer moment pensat i ideat per a la frase, aquesta zona dreta de la graella. D'aquesta manera, a mesura que ha anat avançant el projecte, hi hem anat afegint els altres elements reaprofitant aquest espai. Es pot comprovar, doncs, que la premissa d'intentar fer ús del menor espai possible i d'aquesta manera robar-li'n el mínim possible a la graella (l'element que ja hi era) s'ha tingut present en tot moment.

### 6.3.2.- Disseny de les interaccions

Un cop vistos i justificats tots els elements que conformen aquesta nova interfície, toca parlar, analitzar i justificar totes les interaccions que hem preparat per als usuaris, començant per les interaccions referents a la **graella**.

En primer lloc, dir que evidentment es mantenen les interaccions de què ja disposava en aquesta pantalla abans de l'ampliació amb el nostre mòdul, que són l'**scroll** per a **desplaçar el contingut** de la graella i la **reproducció d'un missatge**. La manera per a reproduir un missatge, doncs, segueix sent fent-hi **clic** sobre l'element de la graella.

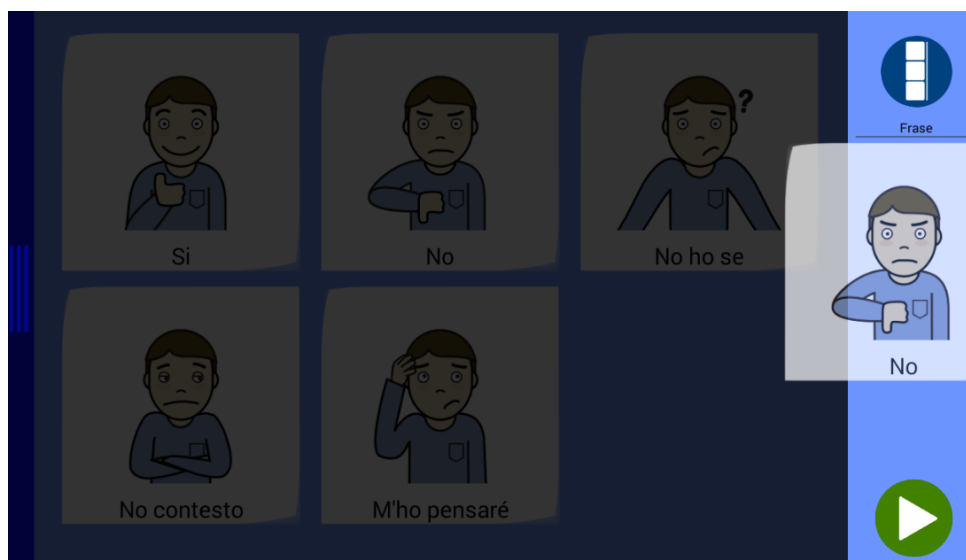
Ara, a més del clic, n'hi trobem una de nova, que es traca del **clic prolongat** (el que es coneix com un long-clic). Així doncs, al fer aquest long-clic sobre un element de la graella, veiem com, per una banda, ens apareix una "còpia" amb menys opacitat del missatge seleccionat i, per altra banda, la graella s'enfosqueix. Veiem a més que si ens movem per la pantalla (mantenint el clic) arrosseguem aquesta còpia del missatge sobre el qual hem fet long-clic, i que, a diferència de la graella, la zona de la frase segueix il·luminada.



Imatge 42: Còpia missatge

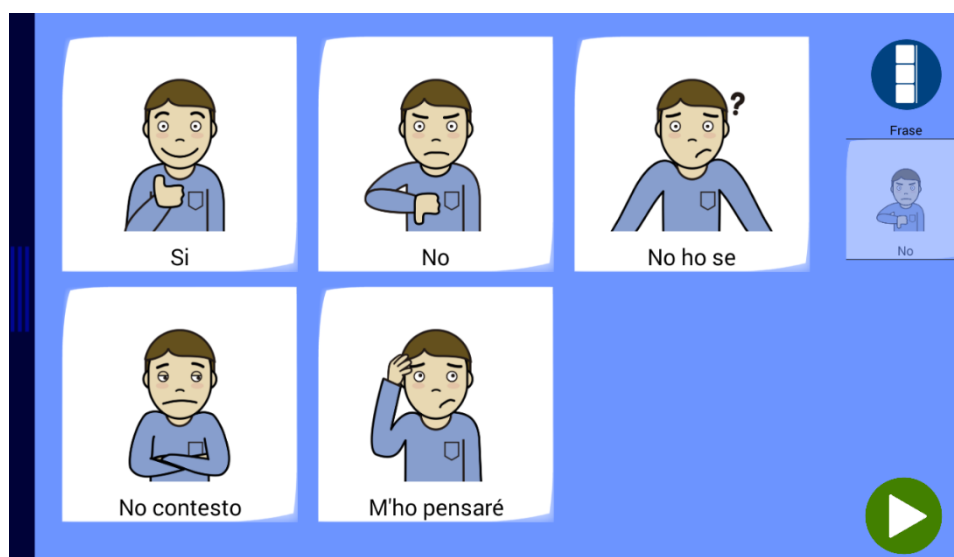
Efectivament, com indica el fet que la frase romangui il·luminada, es traca de la interacció per **afegir missatges** a la frase. A l'arrossegar, doncs, aquest element fins a la zona de la frase i deixant-lo anar, veiem com la frase passa a contenir aquest missatge i la graella torna a quedar il·luminada. Evidentment, deixar anar el missatge sobre la graella no comporta cap acció, motiu pel qual hem il·luminat només la zona de la frase.





Imatge 43: Afegiment missatge frase

No obstant, aquest fet no es produeix immediatament; veiem que en deixar anar el missatge a la frase s'inicia una animació d'aparició del missatge en qüestió, el qual va guanyant opacitat des del 0% fins al 100% amb en un lapse de temps d'un segon. Aquesta animació està pensada per ajudar a veure a l'usuari d'una manera més evident com l'element s'afegeix a la frase.



Imatge 44: Animació aparició progressiva

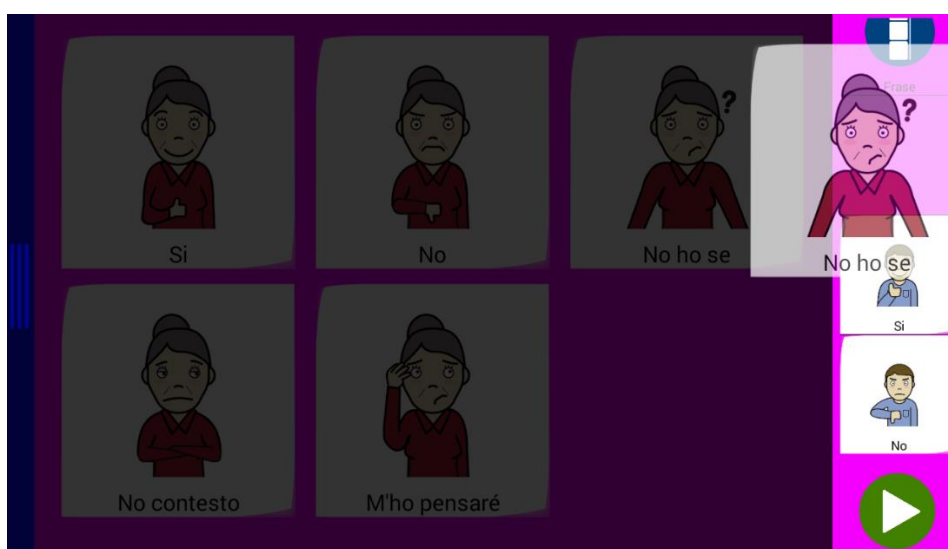
Seguint amb l'afegiment, cal parlar d'un altre escenari bastant freqüent, que és l'afegiment a una posició concreta de la frase; la inserció d'un element. Si arrosseguem un missatge cap a la frase i aquesta ja disposa d'elements, veurem que al desplaçar-nos-hi per sobre hi va apareixent un "forat", el qual representa la ubicació on es col·locarà el missatge en cas de deixar-lo anar en aquell moment.

Evidentment, si ens desplaçem cap a les zones superior o inferior de la frase, es produirà un scroll per a poder seguir "pujant" o "baixant" segons convingui. I de la

mateixa manera que en l'anterior cas, al deixar anar l'element s'iniciarà l'animació d'aparició progressiva en la posició corresponent de la frase.



Imatge 45: Forat frase 1



Imatge 46: Forat frase 2

Com ja comento en l'anàlisi de requeriments, en un primer moment la idea era que l'afegiment del missatge es produís com a conseqüència de la reproducció d'aquest, és a dir, que fent clic en un missatge es reproduïa i després s'afegia a la frase. Aleshores, l'animació de reducció posterior que es produïa fins al lloc del missatge (des d'on s'havia iniciat l'animació prèvia d'ampliació) s'efectuava en aquest cas fins a la posició pertinent a la llista.

Més endavant, però, vam decidir que era millor "deslligar" aquestes dues accions i vam fer que el mecanisme per afegir un missatge a la frase fos amb l'arrossegament. D'aquesta manera, insistim en que les dues accions (reproduir i afegir) romanien independents i la col·locació d'un missatge a la frase es produïa de la manera més intuïtiva possible, agafant l'element i deixant-lo on es desitja.

Vistes les interaccions que es poden dur a terme sobre la graella (tot i que, com hem vist, les accions que permeten també estan bastant relacionades amb la frase) analitzarem les que es permeten fer sobre l'element principal de la dreta, la **frase**, que veurem que bàsicament són les mateixes que les que hem vist de la graella, però amb el propòsit invers.

En primer lloc, destacar que en aquest cas, fent **scroll**, a més de desplaçar el contingut de la frase es poden activar els indicadors de contingut, les dues fletxes situades a la part superior dreta i inferior dreta de la frase respectivament. Dir també que en cas el **clic** sobre un element no comporta cap acció o esdeveniment, a diferència que en la graella. Vam valorar que també fent clic sobre el missatge en la llista aquest es reproduís individualment, però vam concloure que era innecessari. Així doncs, ens centrem en la interacció restant que sí que es permet, que és el **long-clic**.

Aleshores, igual com succeeix en la graella, ens apareix la còpia del missatge de la frase sobre el qual hem fet el long-clic. En canvi, a diferència d'abans, en comptes de quedar una zona enfosquida i una altra il·luminada, veiem com sobre la graella s'hi sobreposa una imatge d'una paperera, i en la frase, l'únic canvi que hi ha és que el missatge seleccionat (no la còpia) també perd bona part de la opacitat, gairebé tota.



Imatge 47: Eliminació missatge

Com podem veure, sovint és important proveir aquest tipus de detalls (zones enfosquides, opacitats d'elements i dels seus homòlegs..) que, tot i ser senzills, ajuden molt i suposen una gran aportació per ajudar a entendre no només el que succeeix en l'aplicació (*feedback*) sinó també el que pot i no pot fer l'usuari, en aquest cas a quina zona i sobre quin element tindrà efecte el fet de deixar anar, aquest *drop*.

Així doncs, efectivament, el que aquest escenari ens indica és que ens estem enduent el missatge de la frase (abans ens n'emportàvem una còpia, per afegir-la) i que deixant-la anar sobre la zona de la graella, és a dir, sobre la paperera, aquest desapareixerà de la frase, serà esborrat.

I així és, deixant-lo anar sobre la paperera (tota la zona de la graella, qualsevol lloc de la zona en vermell), aquesta desapareix i, en la ubicació de la frase on estava col·locat el missatge arrossegat, s'hi inicia l'animació inversa a la que hem vist en el cas de l'afegiment, una animació de desaparició progressiva en què l'element va perdent

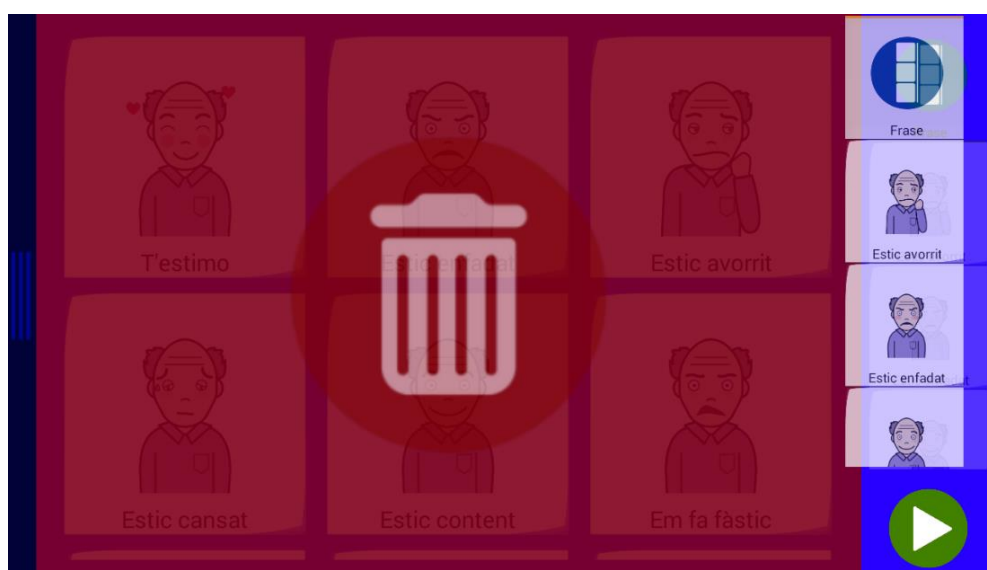
opacitat, del 100% al 0%, al llarg d'un segon, i posteriorment s'elimina i es reubiquen els missatges restants en la frase.



Imatge 48: Animació desaparició progressiva 1

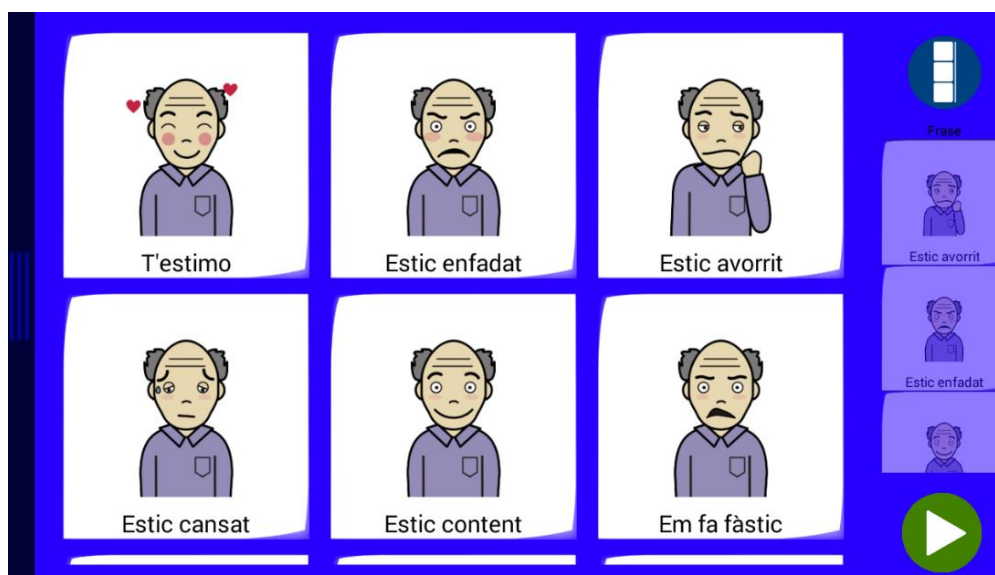
I, per acabar amb les interaccions relatives a la frase, falta parlar sobre les d'un element especial de la llista, que és la ja explicada capçalera, de la qual ara en veurem el funcionament. Com hem dit abans, duent a terme les mateixes interaccions en aquest element, s'accionen els mateixos mecanismes però englobant la totalitat de la frase.

És a dir, fent long-clic sobre l'element de la capçalera, obtenim una còpia de la frase sencera (del contingut de frase que veiem en aquell moment) i veiem com tota la frase perd opacitat. En efecte, estem arrossegant la frase cap a la paperera per esborrar-la sencera.



Imatge 49: Esborrat frase

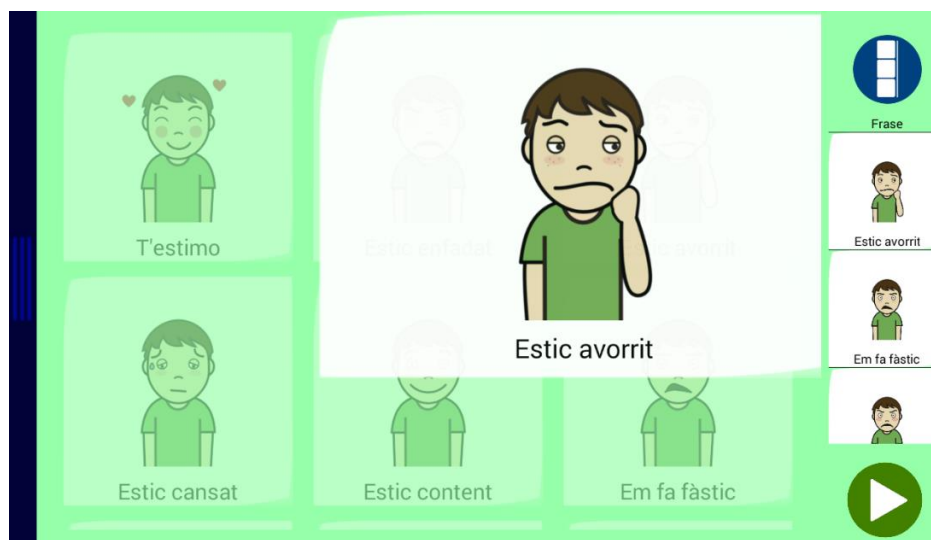
I, si deixem anar aquesta ombra de tota la frase sencera, a banda de desaparèixer la paperera, l'animació de desaparició progressiva s'inicia per tots els missatges exceptuant, evidentment, la capçalera, la qual sempre hi és.



Imatge 50: Animació desaparició progressiva 2

Així doncs, com hem pogut veure, tenim les mateixes interaccions (aplicades a diferents elements) per a dur a terme accions inverses (afegir i esborrar). Pensem que de cara al nostre usuari això suposa un encert, ja que a banda de ser, sota el nostre punt de vista, la interacció més adient per a posar i treure elements d'un espai concret, el fet que ambdues accions es duguin a terme amb el mateix mecanisme és un avantatge i un aspecte positiu de cara a "l'aprenentatge" del funcionament de l'aplicació.

I per acabar amb les interaccions dels nous elements de la interfície, tenim la que potser és més important, sobretot en quant al propòsit o l'acció que desenvolupa. Es tracta del **botó de reproducció** i la reproducció en sí.



Imatge 51: Animació ampliació

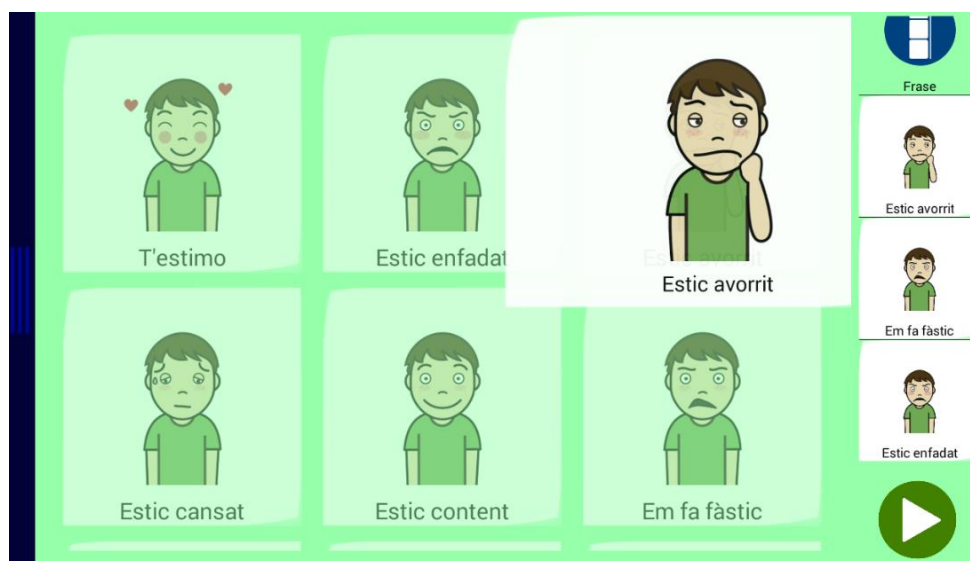
Evidentment, la reproducció de la frase s'inicia fent **clik** en el botó de reproducció. El primer que veiem, doncs, en activar aquest mecanisme, és que s'inicia una animació d'ampliació d'una còpia del primer element de la frase que neix en la seva part superior i que creix i es mou fins a tapar la zona de la graella, i arranca la reproducció.



Imatge 52: Reproducció frase

Aquesta és la principal diferència de la reproducció de la frase respecte a la individual, l'espai que ocupa. Com veiem, a banda de per diferenciar ambdues reproduccions, es deu sobretot al fet que, quedant la frase “destapada”, se'ns permet poder indicar quin element de la frase és el que s'està reproduint. Això ho aconseguim jugant amb les opacitats dels elements; només el que volem remarcar en cada moment té una opacitat del 100%, els altres són translúcids.

Un altre fet destacable que succeeix durant la reproducció dels elements és que el botó de reproducció queda reemplaçat pel de sortir, el qual, fent-hi clic en qualsevol moment, activa de forma prematura l'animació de reducció inversa a la d'ampliació, per la qual la imatge del missatge que s'estigués reproduint es redueix fins a desaparèixer i torna de nou cap a l'extrem **superior** de la frase.



Imatge 53: Animació reducció sortir

Aquest canvi de botó es produeix en el moment d'iniciar-se la reproducció, un cop finalitzada l'animació prèvia. Durant aquest procés (l'animació d'ampliació) seguim veient el botó de reproducció, però, evidentment, tornant a clicar-lo no es produirà cap



efecte nou. D'igual manera, un cop fem clic en el botó de sortir de la reproducció, aquest queda immediatament substituït de nou pel de reproducció, però no tindrà cap efecte fer-hi clic fins que acabi l'animació de reducció.

Un detall interessant és que, si no es fa clic en el botó de sortir i s'espera a que la reproducció de la frase acabi, tindrà lloc la mateixa animació de reducció, en aquest cas, amb l'últim missatge de la frase, i l'animació de reducció es produirà amb l'extrem **inferior** de la frase com a destí.



Imatge 54: Animació reducció finalitzar

El que volem remarcar amb això, és la diferència que hi ha en l'animació final en els dos modes de finalitzar la reproducció. Si l'aturem, l'animació de reducció torna cap amunt, indicant que no s'ha arribat al final de la frase i que per tant, simbolitza una tornada "enrere", una cancel·lació. En canvi, si la deixem acabar, aquesta animació es dirigeix cap a baix, com volent indicar que s'ha completat el "camí", la reproducció, en aquest cas.

Dir també, que per a fer notar el pas d'un missatge a un altre, durant la pròpia reproducció, hem incorporat una petita però necessària animació visual que és com una pampalluga que s'executa entre missatge i missatge per a notar aquest pas.

Per últim, afegir que el **menú desplegable** de l'esquerra per a la selecció de categories, queda desactivat, a més a més de durant la reproducció d'un missatge, en la reproducció de la frase.

## 6.4.- Contextualització de la interfície

Abans de passar a parlar de la implementació, m'agradaria fer una reflexió sobre la interfície i la naturalesa de l'aplicació. Com ja he anat repetint al llarg d'aquesta memòria, l'aplicació ECO, per la qual desenvolupo aquest mòdul, està dirigida a nens autistes, que insisteixo, no tenen desenvolupats de la mateixa manera els sentits de la percepció i la intuïció.

Per tant, és important posar-hi èmfasi en els punts que pertoqui, i aquest (el disseny de la interfície i les interaccions) és, de ben segur, el més important. A mesura que anàvem veient tots els detalls de la interfície, he anat justificant el per què de

cadascuna i he descrit animacions o detalls que, sovint, ajuden a comprendre un procés o un fet que té lloc en l'aplicació, el qual, sense ser-hi, o sent menys explícits, segurament ja s'hauria entès.

Així doncs, m'agradaria fer llista i repassar tot aquest seguit de factors que representen aquesta diferència del que seria una interfície per a usuaris comuns respecte a la que és la nostra, i que al final són els elements que aporten valor al nostre projecte. Podríem parlar-ne en quatre eixos:

- El **temps** de les animacions. Tant les d'aparició i de desaparició progressiva dels missatges en la frase com les d'ampliació i reducció prèvies i posteriors a les reproduccions individuals i "col·lectives" tenen una durada d'un segon. Pensem que aquest valor elevat de temps ajuda a l'usuari pel que fa a la comprensió de tot el que succeeix en l'aplicació.
- La **mida** dels elements. Tant els elements de la frase com els de la graella, i també el botó de reproducció, tenen unes dimensions considerables. De nou, aquest fet també aporta el seu granet de sorra per a una assimilació més fàcil i efectiva de la interfície.
- L'**estil** dels elements. Tots són elements extremadament senzills, sense floritures, fins i tot infantils. Evidentment, l'aplicació ECO va dirigida a nens, però nens autistes. Que tots els components siguin el més senzills possible en quant al disseny ajuda a evitar confusions.
- El caràcter **intuïtiu** de les interaccions. Pensem que tant les interaccions que ja hi havia com les noves (clic i long-clic + arrossegament) representen de forma clara i evident les accions i efectes que comporten.



## 7.- IMPLEMENTACIÓ

I per acabar els tres apartats centrals d'aquesta memòria, trobem la implementació. En un projecte informàtic d'aquest tipus, és a dir, de desenvolupament, no de recerca, els tres principals blocs de què es compona l'informe o memòria són aquests: especificació, disseny i implementació. I ho són perquè són els que ens porten de la definició tècnica de problema al producte final en sí, en aquest cas el nostre mòdul per a l'aplicació ECO.

Així doncs, en aquest punt definirem de quina manera hem implementat els diferents elements i interaccions que conformen la interfície, vistos i descrits ja en l'anterior punt; el disseny. Al llarg d'aquest apartat aniran apareixent conceptes tècnics relatius a la programació en Android que poden ser desconeguts, dels quals en donarem, en el mateix moment, una breu definició i explicació en la qual, sense entrar-hi en detalls, esperem deixar-los clars.

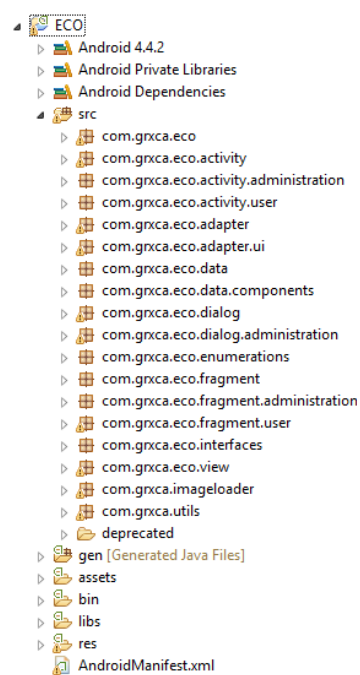
No obstant, però, és important, primer de tot, definir en quina situació ens trobem a nivell de programació. És a dir, hem comentat que per al desenvolupament del projecte fem ús d'un IDE com és l'eclipse i que aquest tipus de programes treballen amb 'projectes', proveint una organització estructural de totes les classes i recursos que componen cada projecte. Per tant, penso que és interessant veure com està organitzat el projecte ECO en l'IDE i veure com podem introduir-nos-hi amb el nostre mòdul.

### 7.1.- Marc contextual

Portem bona part de la memòria parlant d'aquesta diferenciació que fa ECO pel que fa als àmbits; zona d'usuari i zona d'administrador. Donant un cop d'ull a l'estructuració en classes i components de l'aplicació (imatge de la dreta) podem veure que, sovint, en segons quins elements, es manté aquesta dualitat.

Com veiem en la imatge lateral (l'estructuració del projecte en l'IDE), trobem els diferents *packages* de què es compona el projecte i en alguns hi podem veure aquesta classificació en '*administration*' i '*user*', com és el cas de les *activities* i els *fragments*.

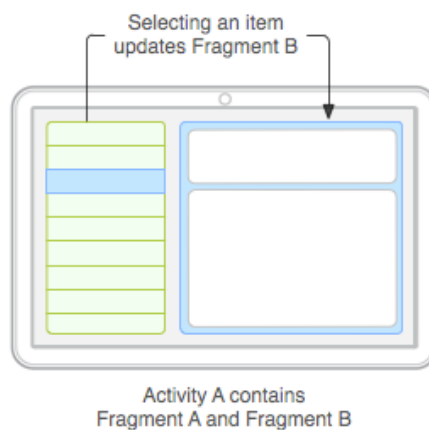
Començo parlant d'això, ja que, per veure "per on" podem entrar a desenvolupar el nostre mòdul o ampliació, són aquests dos packages els que s'hi veuen involucrats. Pel que fa a les **activities**, que són els components d'Android que representen les pantalles en sí, i que es diferencien de la resta de components i classes per disposar d'una part gràfica a més de la part lògica, ens hem de centrar en l'activity de la secció d'usuari, la que seria la ja vista pantalla de selecció de categories.



Imatge 55: Estructura projecte IDE

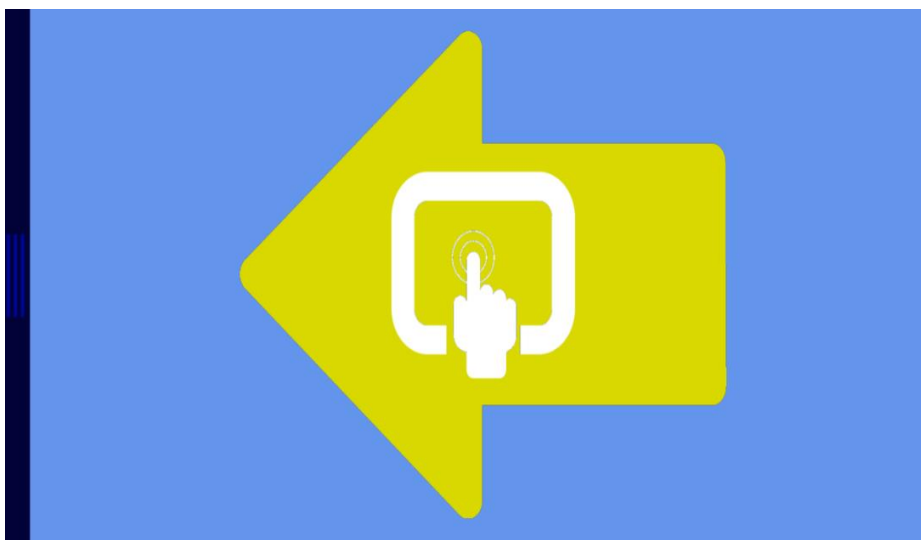
Aquesta pantalla, doncs, està programada per mitjà d'un **fragment**, el que en Android es podria definir com una **porció d'una activity**. D'aquesta manera, es pot aconseguir tenir en una mateixa activity més d'un fragment, tots ells independents (cadascun amb el seu propi cicle de vida, els seus propis *events*) però dependents de l'activity *host*.

En la imatge de la dreta podem veure una representació del típic exemple de combinació de dos fragments en una activity, i que a efectes pràctics és molt similar al que implementa ECO per a la selecció de categories i l'ompliment del seu contingut en la graella de la seva pantalla corresponent.



Imatge 56: Fragments

Així doncs, entrant en la part que ens interessa, quan seleccionem un usuari en el menú principal d'ECO, accedim a la UserSectionActivity (secció 'user' de les activities), la qual és l'activity host del fragment. El que disposa concretament aquesta activity en el seu layout és un *SlidingMenu* (aquest llistat que despleguem a la part esquerra de la pantalla per a seleccionar la categoria) i un fragment, que d'inici ocupa la totalitat de la pantalla i simplement mostra una imatge amb la fletxa que indica la ubicació d'aquest *SlidingMenu*, en un principi, "amagat".



Imatge 57: Fragment inicial

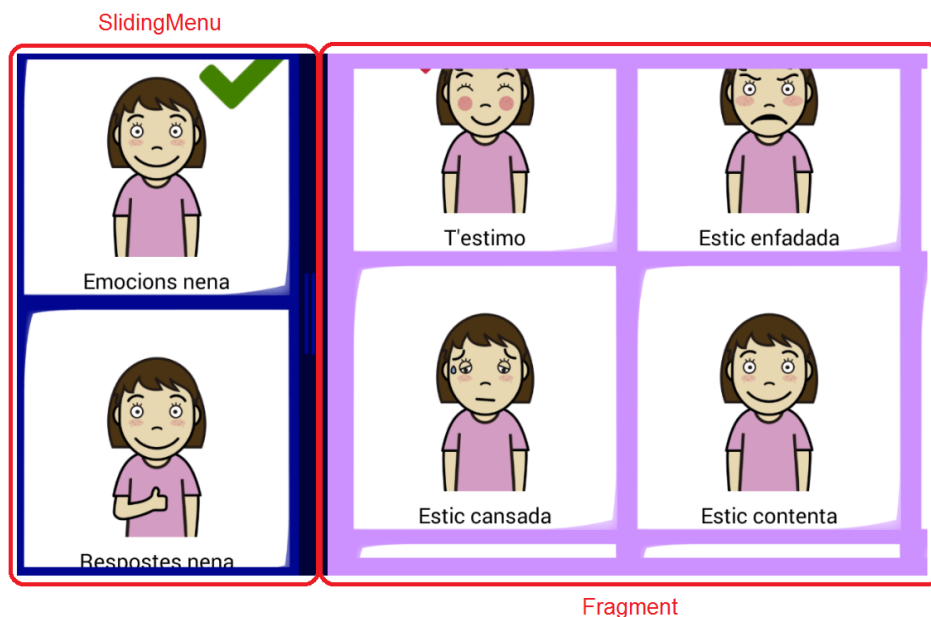
Aquesta imatge de la fletxa ja és un fragment, que ara per ara simplement conté una imatge però més endavant aquesta imatge serà substituïda per un fragment més elaborat, quan seleccionem una categoria.

Al seleccionar, doncs, una categoria, el gestor de fragments d'aquesta activity inicia una transacció en la qual es reemplaça aquesta imatge pel fragment ja "omplert" amb la categoria seleccionada:

```
getFragmentManager()
    .beginTransaction()
    .replace(R.id.user_section_body_layout, mCategoryFragmentP)
    .commit();
```

Imatge 58: Transacció fragments

En fer-ho, podem veure una disposició de la pantalla més similar a la que hem vist en l'exemple de la pàgina anterior. En comptes de tenir 2 fragments, tenim un SlidingMenu i un fragment, però el comportament és molt similar.



Imatge 59: El nostre fragment

D'aquesta manera, ja tenim localitzat l'element que al cap i a la fi és el que hem de modificar o ampliar per tal d'implementar totes les novetats vistes en l'anàlisi de requeriments.

Analitzant aquest fragment (CategoryFragment.java, secció 'user' dels fragments), veiem que simplement disposa en el seu layout (el layout és l'element que defineix la interfície d'una activity i en aquest cas d'un fragment; la seva estructura) dos elements: un GridView, la graella que s'omple amb els diferents missatges de la categoria en qüestió, i un ImageView que es col·loca a sobre de la graella i que es fa visible només quan seleccionem un missatge, quan, acompanyat de l'animació d'ampliació, "s'omple" amb la imatge del missatge seleccionat.

Veiem, doncs, que tenim bastanta feina per endavant. La manera en la que incorporarem el nostre mòdul serà ampliant aquest CategoryFragment.java.

Abans de passar a veure en detall la implementació d'aquesta ampliació, però, definirem dos conceptes més que s'afegeixen al projecte d'ECO en forma de llibreries externes. El primer és aquest SlidingMenú del qual ja n'hem parlat al veure l'estructuració de l'activity, la llibreria del qual permet construir aquest tipus de menús "lliscants" en forma de llista i incorpora tot un seguit d'events per a veure què hi té lloc (moviment, selecció d'un element..).

I el segon és un altre component similar que en aquest cas s'utilitza per a fer el menú de selecció de l'usuari; l'activity prèvia a la UserSectionActivity. Es tracta del ViewPager, una altra llibreria similar en quant a l'element que incorpora, en aquest cas un menú d'objectes amb desplaçament horitzontal i que sempre és visible.

## 7.2.- Implementació de la interfície

Un cop ja vistos a nivell de disseny tots els components i les interaccions de la nostra interfície i havent definit de quina manera incorporarem el nostre mòdul en l'aplicació ECO, és torn de veure com hem dut a terme tota la implementació.

Per a fer-ho, seguirem amb el mateix patró de desenvolupament del punt anterior, i en primer lloc parlarem de la implementació dels diferents elements i components i després passarem a descriure la implementació de les interaccions.

### 7.2.1.- Implementació dels components

D'igual forma que en el disseny, començarem parlant de la **frase**, aquest element bàsic incorporat en l'aplicació. Si ens fixem en les diferents justificacions fetes tant en l'anàlisi de requeriments com en el disseny sobre la seva col·locació i sobretot orientació i funcionament (vertical i lògica *FIFO*), ràpidament veiem que la millor opció per a implementar aquest element és mitjançant un **ListView**, que és un element que presenta els seus components en forma de llista, i en sí no és més que un conjunt de vistes.

Aquests components, els **missatges**, estan implementats en l'aplicació ECO com a objectes de la classe **MessageUI**, la qual implementa tota la seva informació (nom, imatge, so, vídeo...). Tots, excepte aquest primer element; la **capçalera**, que per a implementar-lo aprofitem una especialització d'aquesta classe anomenada **FakeMessageUI**. Com el propi nom indica, es tracta simplement d'una especialització per a representar elements tal com si fossin missatges però sense haver d'accedir a la persistència (base de dades) per a crear-lo.



Imatge 60: Components interfície 1

Un altre punt positiu conseqüència d'implementar la frase mitjançant un ListView és el fet que es podrà aprofitar la mateixa classe *adapter* que utilitza el propi **GridView** (que ve a ser el mateix que el ListView però presentant els seus elements en una graella), el **ComponentUIListAdapter**, una extensió de l'**ArrayAdapter**. En Android, un adapter

és l'element pont entre un objecte de dades i un conjunt de vistes, és a dir, l'encarregat de generar les vistes (en el nostre cas en el GridView i el ListView). Aquest objecte de dades és típicament un array amb els identificadors dels objectes complexos (en el nostre cas els MessageUI i el FakeMessageUI) amb què volem omplir les nostres vistes.

Havent vist la frase, cal centrar-se en els elements restants de la interfície, les imatges **indicadores de contingut** i els **botons de reproducció i sortida** d'aquesta.

Les primeres, evidentment, s'implementen mitjançant dos **ImageView**, un element que simplement presenta la imatge que se li indiqui. Pel que fa a la seva col·locació, les situem als extrems superior i inferior dret, a sobre del ListView de la frase.

Pel que fa al botons de reproducció i sortida, es tracta de dos ImageView més, que en aquest cas, els col·loquem a sota del ListView, l'un a sobre de l'altre (el de sortida només es visible quan es duu a terme la reproducció de la frase). En el cas d'aquests dos botons, hauríem pogut implementar-los mitjançant l'element Button, que semblaria més adequat. No obstant, vam optar per l'ImageView per qüestió de simplicitat.



Imatge 61: Components interfície 2

Per acabar de veure aquest apartat referent a la implementació dels components de la nova interfície, cal acabar de veure com es construeix tota aquesta estructura, el que abans hem dit que era un layout, i acabar de veure alguns components més. De nou, per ajudar a veure la nostra incidència en l'aplicació, presentarem primer l'estructura inicial:

Abans de la nostra incorporació, doncs, el layout del fragment era simplement un **FrameLayout**, que és un tipus de layout amb la característica principal que va apilant els elements que se li van afegint, en format LIFO (el funcionament d'una pila). El primer element afegit queda a sota i els següents s'hi van col·locant al damunt.

Imatge 62: Disposició inicial layout

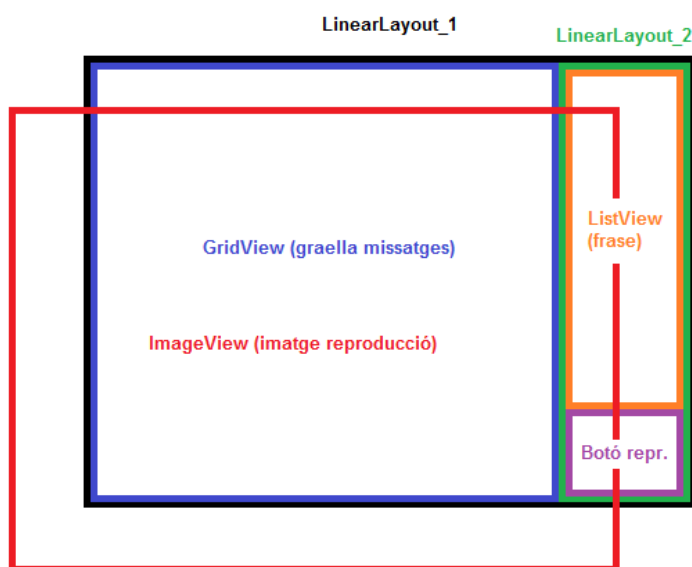


D'aquesta manera, tenim per una banda el GridView dels missatges de la categoria, i, a sobre, l'ImageView a "omplir" quan es reproduïx un d'aquests missatges.

Ara, però, la cosa és una mica més complexa; a mesura que vaig anar afegint-hi elements, vaig anar modificant i incorporant diferents layouts fins que al final l'esquema resultant és digne d'analitzar-lo.

En primer lloc, vam veure que necessitàvem un espai a la dreta per a la frase; el ListView. I també, a sota d'aquesta, un espai per al botó de reproducció. De manera que el que abans era un FrameLayout amb un GridView i un ImageView a sobre va passar a ser el un FrameLayout amb un **LinearLayout** (un tipus de layout que permet la col·locació d'elements al mateix nivell, en el qual cal indicar, per mitjà de regles, si és al costat, a sobre, a sota, etc) i a sobre l'ImageView.

A la seva vegada, aquest LinearLayout, el qual anomenarem LinearLayout\_1, contenia el GridView i a la seva dreta un altre LinearLayout, LinearLayout2, el qual tenia la llista i el botó de reproducció a sota.

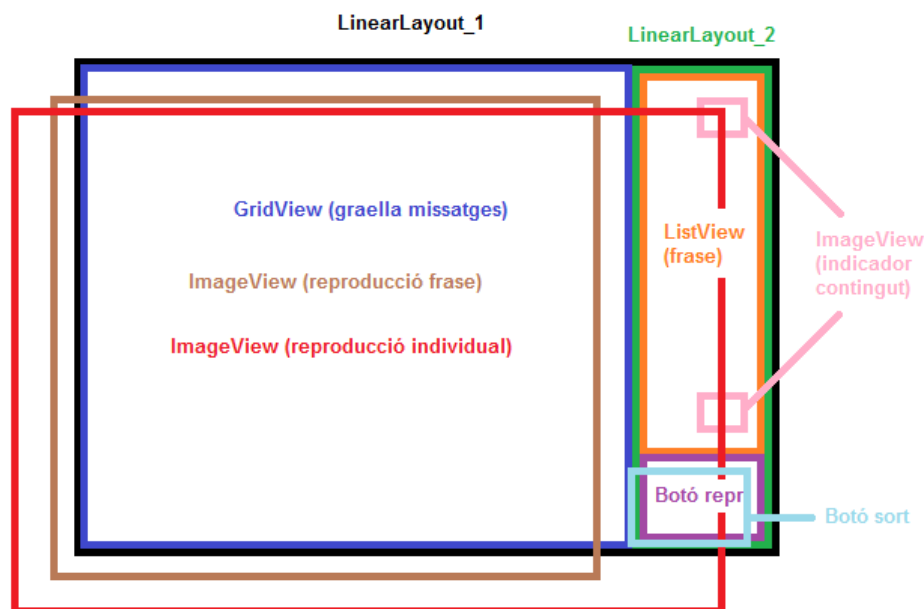


*Imatge 63: Primera remodelació layout*

Més endavant, a causa de dues noves modificacions, la composició es va tornar encara més complexa. En primer lloc, vaig prendre la decisió d'utilitzar una nova imatge sobreposada diferent per a la reproducció de la frase pels motius ja esmentats en el disseny, i, en segon lloc, vaig incorporar les dues imatges indicadores del contingut no visible de la frase i també el botó de sortir de la reproducció.

Això va comportar, per una banda, que on abans hi teníem simplement el GridView ara hi tinguéssim un FrameLayout (FrameLayout\_2) amb el GridView i un ImageView a sobre (per a la reproducció de la frase). Per altra banda, on abans hi teníem la frase ara hi tenim un FrameLayout (FrameLayout\_3) amb el ListView i els dos ImageView a sobre (les fletxes indicadores), i on hi teníem el botó de reproducció ara hi tenim un altre FrameLayout (FrameLayout\_4) amb els dos ImageView corresponents als dos botons.

D'aquesta manera, la composició final va passar a ser la següent:



Imatge 64: Disposició final layout

### 7.2.2.- Implementació de les interaccions

Després de veure quines són les nostres peces, anem a veure quins moviments podem fer-hi. Igual que en el disseny, començarem per la **graella**.

#### **Interaccions amb la graella:**

Com ja diem en el disseny, tot el que ja es podia fer abans es segueix podent fer, no hem alterat “el curs” d’ECO en aquest sentit, de manera que fent clic en un missatge, aquest es reproduïx.

Aquest tipus d’events es detecten per mitjà dels **listeners**, que al capdavant són interfícies de la classe View (tant el ListView, com el GridView i l’ImageView són extensions d’aquesta classe, de manera que també disposen d’aquest element) els quals inclouen un mètode de *callback* en el qual s’hi defineix el procediment a seguir en cas que es produeixi l’event determinat.

Així doncs, el GridView incorpora un **onItemClickListener**, dins el qual hi ha definit el procés d’inici de la reproducció d’un missatge concret (aquest *ítem*). Quan parlem de reproduccions veurem com és aquest procés amb més detall.

Ara, doncs, a més d’aquest listener per al clic, n’hi incorporarem un altre; el **onItemLongClickListener**, el qual detecta quan es produeix un clic prolongat sobre un missatge del GridView. Com diem en el disseny, l’acció immediata que això comporta és la creació d’aquesta còpia del missatge relacionat amb menys opacitat, la qual creem mitjançant la classe DragShadowBuilder, i activem el flag (un indicador de mode, que es pot implementar amb un integer mateix) d’afegiment, que com veurem necessitarem per a detectar si es tracta d’un afegiment a la frase o no.





Imatge 65: Shadow missatge graella

Com hem vist també en el disseny, s'activa també un filtre de fosc per a tota la zona de la graella mentre que la zona de la frase segueix "il·luminada", la qual cosa ens indica que aquest element shadow que ens apareix i podem arrossegar per tota la pantalla, molt probablement tindrà un efecte si es deixa anar en la zona de la frase.

Aquí, doncs, hi intervé un nou listener, en aquest cas de l'objecte `ListView`. Aquest nou listener és el **`onDragListener`**, el qual detecta qualsevol arrossegament per a l'objecte en concret. A més a més, dins d'aquest listener podem discernir en diferents accions relacionades amb aquest event, i segons quina acció es tracti es durà a terme un procediment o un altre.

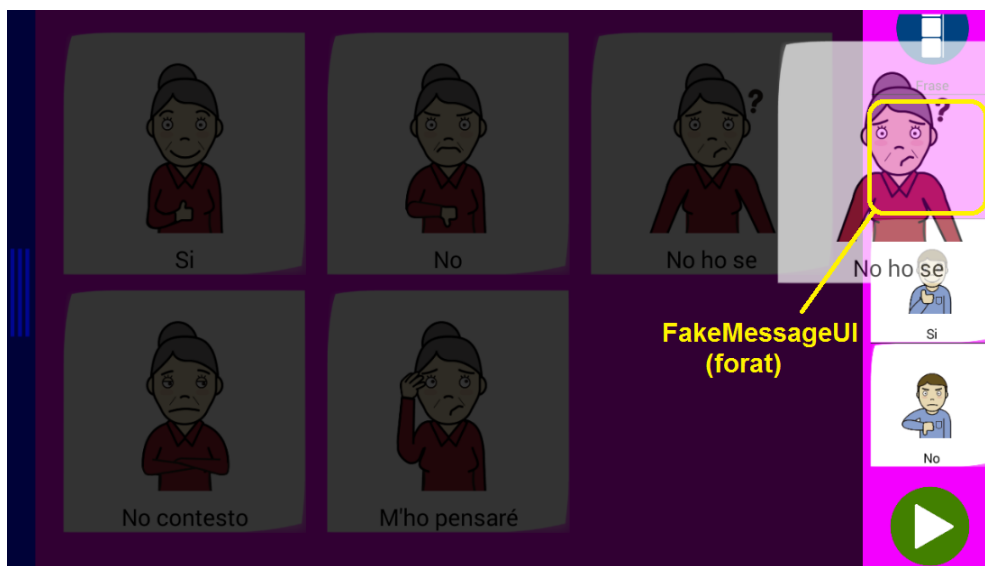
Per al nostre cas, les accions que hem diferenciat i actuat en conseqüència són, entrada a la zona de l'objecte en concret, moviment per la zona (es detecta cada vegada que varia en un píxel la posició), deixar anar en un lloc de la zona, sortida de la zona i finalització de l'arrossegament.

Així doncs, pel que fa a l'afegiment de missatges en la frase, podem diferenciar dos escenaris diferents, afegiment en una frase buida o afegiment en una frase que ja contingui elements. En aquest cas començarem parlant del cas complex, el d'afegiment en una frase no buida.

Com que volem que, al desplaçar-nos, per la frase s'hi vagi movent un "forat" que representi la posició on s'ubicaria l'element en cas de deixar-lo anar en aquell moment, la "trampa" que fem és, en detectar que un element entra en la zona de la frase i, efectivament, el flag del mode afegir està activat, afegim un altre **`FakeMessageUI invisible`** al final.

D'aquesta manera, ja tenim el "forat" creat, i a partir d'aleshores, cada vegada que el listener detecta un desplaçament per la zona de la frase, es calcula la posició sobre la qual està situat el missatge en aquell moment a partir de les coordenades (píxels) i el forat passa a ocupar aquella posició. Per a fer-ho, esborrem el `FakeMessageUI` i el tornem a afegir a la nova posició.





Imatge 66: Forat frase

Aleshores, quan el listener detecta un drop, és a dir, que es deixar anar el missatge sobre algun punt de la frase, s'elimina aquest FakeMessageUI i s'insereix el nou missatge (recordem que cada vegada que afegim, el que fem és crear un nou missatge, ja que volem que en una frase hi pugui haver missatges repetits), ara sí, ja visible, i desactiva el mode afegir, tornant a tenir també la graella il·luminada.

Per aconseguir aquest fet, que un element del ListView sigui o no visible, activem un flag de l'adapter i li indiquem quina és la posició que ha de mantenir invisible. Aleshores, com que en l'adapter cada vegada que s'hi produeix un canvi (afegiment, inserció, esborrat) torna a "pintar" les vistes del seu objecte, si en algun moment veu que tenim el mode invisible activat i aquella és la posició que volem que sigui invisible, simplement li activa la invisibilitat. Més endavant en veurem un altre exemple.

En canvi, si no es produeix el drop i se surt de la zona de la frase, s'elimina aquest FakeMessageUI, ja que en aquest cas no hi volem veure cap forat, en la frase. Pel que fa a l'acció restant, finalització de l'arrossegament, en aquest cas simplement desactivem el mode afegir, ja que s'haurà fet el drop en alguna zona fora de la frase, i per tant, sense cap conseqüència, i traiem la foscor a la graella.

Afegir també que, si en l'acció de desplaçament, el listener detecta que ens situem en els extrems superior o inferior de la frase, es produeix un desplaçament per a seguir mostrant contingut (si n'hi ha més). Pel que fa al cas senzill, afegir en una frase buida, només ens interessa l'acció del drop, en la qual simplement s'afegeix el missatge, evidentment en la primera posició, just després de la capçalera.

### Interaccions amb la frase:

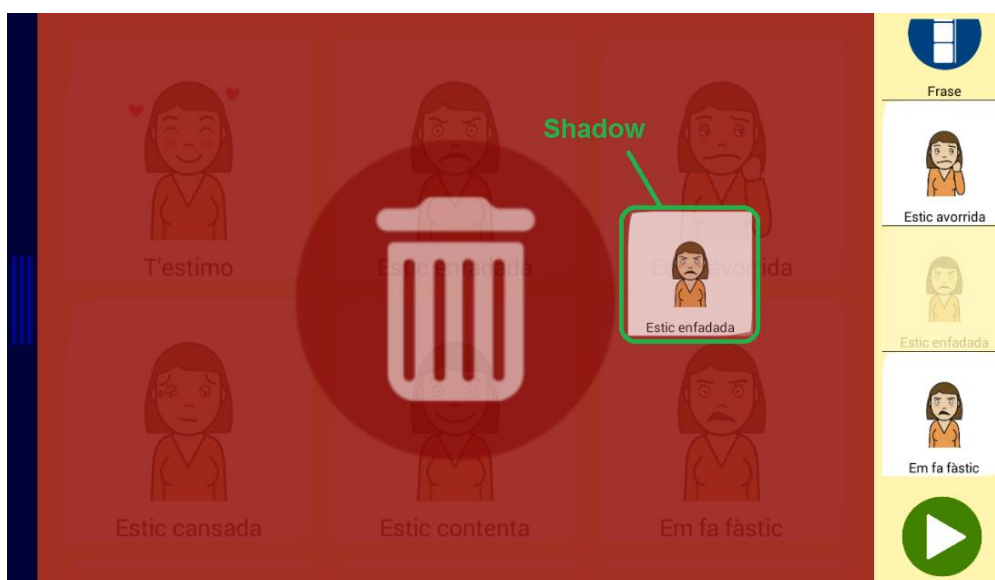
Passem a parlar de les interaccions de la frase. Com diem en el disseny, tenim dues interaccions destacables que poden comportar canvis sobre aquest element. En primer lloc, l'**scroll**, que com ja hem explicat, si en algun moment queda contingut de la frase "amagat" tant per dalt com per baix, s'activa l'indicador de contingut corresponent (o els dos).

Per a fer això, tenim un listener en el ListView de la frase que és l'**onScrollListener**, el qual detecta cada vegada que es produeix un desplaçament en la llista, i si veu que el primer element (sense tenir en compte la capçalera) o l'últim no són visibles, el que

fem és tornar visible l'ImageView corresponent, ja que evidentment els inicialitzem invisibles. Si pel contrari, detecta que el primer o l'últim és visible, restableix aquesta invisibilitat a l'ImageView pertinent.

I a banda d'aquest listener per l'scroll, evidentment també tenim el mateix listener vist abans en la graella, pel cas del long-clic. En aquest cas, quan l'onItemLongClickListener del ListView detecta un clic prolongat sobre algun dels seus elements, a més de fer-ne aparèixer la còpia amb el DragShadowBuilder, torna visible la imatge de la paperera que, com ja hem vist en el punt anterior, “tapa” (ja que és parcialment translúcida) tota la graella, i activa el flag del mode eliminar.

A més a més, juguem amb el component alpha de la vista en el ListView del missatge seleccionat i el posem a un 50%, restant-li així opacitat a aquell missatge per a indicar que amb aquest arrossegament ens l'estem enduent realment, no com en el cas de l'afegiment, que sempre ens emportem una còpia.



*Imatge 67: Shadow missatge frase*

No obstant, en el onDragListener que també necessitem en el GridView per a poder completar les eliminacions, no necessitem diferenciar tantes accions diferents, ja que en aquest cas tot és més senzill. Simplement detectem el drop, en el procediment del qual procedim a eliminar el missatge corresponent en l'adapter.

Això ho podem fer amb certa facilitat, donat que, en aquest mateix listener, podem obtenir la vista de l'objecte que arrosseguem (aquest shadow), i com que l'adapter inclou un tag (una etiqueta) en cada vista a partir del qual podem recuperar l'objecte MessageUI (que recordem que és el component de què es compona la llista), aquesta eliminació es pot efectuar fàcilment.

L'altra acció que necessitem poder diferenciar es la finalització de l'arrossegament, per la qual podem saber si aquest ha finalitzat en algun lloc fora de la graella (i, per tant, no té conseqüències), i aleshores desactivar el flag del mode eliminar, acció que també es duu a terme en el cas del drop, a més a més de fer desaparèixer la imatge de la paperera i que el missatge al qual li havíem “pres” opacitat la recuperi.

I per acabar amb la frase, recordar que tenim la possibilitat d'esborrar-la sencera duent a terme el mateix procés que amb un missatge sol, però, amb l'element capçalera. En l'onItemLongClick listener, si l'element seleccionat és el primer creem un shadow de

tota la frase, i en l'acció drop de l'onDragListener de la graella fem un clear() de l'adapter.

### La reproducció:

Per a la reproducció de la frase (i aquí veurem també en més detall com funciona la reproducció individual) la interacció que necessitem és la del clic en el botó de reproducció, de manera que incloem un onClickListener en l'ImageView corresponent.

El primer que fem en aquest cas, doncs, és “donar-li” el primer MessageUI de la frase (sense comptar la capçalera) a la imatge fins ara invisible que tenim col·locada tapant tota la zona del GridView. Aquesta imatge, no obstant, no és un ben bé un ImageView, sinó que és un objecte d'una classe extensió d'ImageView, anomenada MessageView, en la qual hi tenim les diferents funcions relatives a la reproducció.

Un cop finalitzada l'animació d'ampliació (la implementació de la qual detallarem més endavant) s'inicia la reproducció d'aquest MessageView amb un startMediaPlayBack (funció d'aquesta classe). Recordem que la reproducció d'un element consisteix en imatge(temps configurable) + vídeo(temps vídeo) + so(temps so), i que l'únic que és “obligatori” és la imatge.

Un cop finalitzada la reproducció del primer MessageUI de la frase, es genera un event que el propi fragment detecta, i si identifica l'acció en concret com un STOP de reproducció (la fi, en aquest cas), es mira si hi ha més missatges a reproduir en la frase.

En cas afirmatiu, es procedeix a “omplir” de nou el MessageView i a iniciar l'animació de transició entre missatges, la qual, un cop acaba, dona lloc al següent startMediaPlayBack, i així successivament. Si pel contrari, veiem que el que acaba de reproduir-se era l'últim missatge, o bé s'ha activat la sortida forçada de la reproducció, es procedeix a l'animació de reducció.

Com hem comentat en el disseny, un cop s'inicia la reproducció del primer element de la frase, el botó de reproduir queda substituït pel de sortir (el tornem visible i l'habilitem), al qual també li incloem un onClickListener, que crida la funció stopMediaPlayBack del MessageView i genera també un event d'STOP en el fragment.

En aquest context, destacar també el fet que, de nou, utilitzem l'adapter per anar jugant amb la opacitat dels missatges de la frase, la qual, recordem, queda destapada. Així doncs, en iniciar-se la reproducció activem el flag de l'adapter relatiu al mode reproducció, i en cada “pas” li indiquem quina és la posició de la vista que volem opaca.

En cada canvi, a més, forcem l'adapter a que hagi de tornar a pintar totes les vistes, i si detecta que el mode reproducció està activat i que una posició ha de tenir opacitat 100%, li posa. Si no, li posa el paràmetre alpha (la opacitat) al 50%.

Un cop vist tot el funcionament que hi ha darrere de la reproducció d'una frase, el de la reproducció individual ens semblarà d'allò més senzill. En aquest cas, en fer clic en un dels missatges del GridView, posteriorment a l'animació d'ampliació que es produeix, s'executa la funció startMediaPlayBack al MessageView corresponent (que en aquest cas ocupa tota la pantalla), i quan el fragment detecta l'STOP, o bé per finalització natural o bé per finalització forçada, es procedeix a l'animació de reducció.

Per a sortir de l'animació individual, n'hi ha prou amb fer un clic prolongat sobre la imatge gran on s'està duent a terme la reproducció. Per això, cal incloure un

onLongClickListener a aquest MessageView, el qual executa un stopMediaPlayerBack i aquest event d'STOP.

### **Les animacions:**

I per acabar de veure la implementació de les diferents interaccions en aquesta nova interfície, parlarem de les animacions. En el disseny les anàvem comentant a mesura que tenien lloc en les interaccions que s'anaven detallant, però en aquest punt, al seguir totes un mateix "patró" d'implementació, les agrupem en aquest subapartat final.

Les diferents animacions que tenim en la nostra interfície són les d'ampliació i reducció tant pel cas de reproducció individual com de tota la frase, i les d'aparició i desaparició progressiva que tenen lloc just abans i just després de l'afegiment i la eliminació, respectivament. A més, tenim l'animació de transició entre missatges en la reproducció.

En totes elles la implementació es molt similar; ens ajudem en cada cas d'un objecte de la classe AnimatorSet, al qual se li poden afegir diferents propietats. Aquestes propietats poden ser tant translacions (en les quals s'hi indica l'objecte, l'origen i el destí), escalats (objecte, mida inicial i mida final, o bé objecte i factor d'escalat), durada (temps en ms), canvis d'opacitat (objecte, opacitat inicial, opacitat final), etc.

Així doncs, en cada cas (en cada animació) es defineix un objecte AnimatorSet i se li incorporen les propietats necessàries:

- Ampliació reproducció individual:
  - Translació del MessageView des de l'origen del missatge en qüestió en la graella fins als extrems de la pantalla.
  - Escalat del MessageView des de 0 fins a ocupar tota la pantalla.
  - Canvi d'opacitat del MessageView des del 85% al 100%.
  - Canvi d'opacitat del GridView, ListView i ImageView del botó de reproducció des del 100% al 0%.
- Ampliació reproducció frase:
  - Translació del MessageView des de l'extrem superior de la frase fins als extrems de la pantalla.
  - Escalat del MessageView des de 0 fins a tapar tota la graella.
  - Canvi d'opacitat del MessageView des del 85% al 100%.
  - Canvi d'opacitat del GridView des del 100% al 0%.
- Reducció reproducció individual:
  - Translació del MessageView des dels extrems de la pantalla fins a l'origen del missatge en qüestió.
  - Escalat del MessageView fins a 0.
  - Canvi d'opacitat del MessageView des del 100% al 85%.
  - Canvi d'opacitat del GridView, ListView i ImageView del botó de reproducció des del 0% al 100%.
- Reducció prematura reproducció frase:
  - Translació del MessageView des dels extrems de la pantalla fins a l'extrem superior de la frase.
  - Escalat del MessageView fins a 0.
  - Canvi d'opacitat del MessageView des del 100% al 85%.
  - Canvi d'opacitat del GridView des del 0% al 100%.

- Reducció natural reproducció frase:
  - Translació del MessageView des dels extrems de la pantalla fins a l'extrem inferior de la frase.
  - Escalat del MessageView fins a 0.
  - Canvi d'opacitat del MessageView des del 100% al 85%.
  - Canvi d'opacitat del GridView des del 0% al 100%.
- Aparició progressiva:
  - Canvi d'opacitat del MessageUI en qüestió en la frase del 0% al 100%.
- Desaparició progressiva:
  - Canvi d'opacitat del MessageUI en qüestió en la frase del 100% al 0%.
- Transició entre missatges en la reproducció:
  - Canvi d'opacitat del MessageView des del 50% al 100%.

I totes aquestes animacions, amb una durada d'un segon (excepte la de transició). A més a més, a aquest AnimatorSet se li poden afegir diferents listeners, com ara d'inici, de cancel·lació, de finalització, etc. El que a nosaltres ens interessa és el de finalització, ja que duem a terme les accions pertinents un cop acabades les animacions.

Per acabar, destacar el fet, gairebé com a curiositat, que totes les animacions estan implementades en el fragment a excepció de la d'aparició progressiva. Això es deu a que, com que s'ha de realitzar l'animació abans d'afegir pròpiament el MessageUI al ListView, en el moment de l'animació, encara no podem disposar d'ell en el fragment, ja que no hi és, i és l'objecte que necessitem animar.

Aleshores, de manera semblant a quan volíem que un dels MessageUI de la frase fos invisible o tingués una opacitat diferent, hem hagut d'afegir un nou flag a l'adapter corresponent al mode afegiment, i quan va pintant les vistes, si ens trobem en mode afegiment i la posició que està pintant és la que ens interessa, realitza l'animació sobre aquella vista, el MessageUI que afegim.

Així doncs, a més d'ampliar el contingut d'aquest CategoryFragment, també hem necessitat poder modificar (o ampliar) aquest adapter, que recordem és el ComponentUIListAdapter.

### 7.3.- Especificació, disseny i implementació

Ja per acabar, doncs, aquesta part central del projecte, en la qual hem passat de veure quines opcions permetia dur a terme aquesta pantalla i contrastar-hi els nous requeriments a definir com hem implementat la nostra solució, m'agradaria remarcar un fet. Si ens hi fixem, tant en l'especificació com en el disseny i també en la implementació hem fet aquesta "separació" conceptual que consisteix en **estructura i comportament**.

En l'especificació, la definició del problema a resoldre, hem separat el que és el model estàtic, el conceptual (conceptes i relacions) del que són els casos d'ús, el que seria el model dinàmic.

En el disseny, quan parlàvem de definir la solució a partir dels elements descrits en l'especificació, hem separat el model de components (estructura) del de

comportament. I també, en la interfície, hem detallat primer els diferents elements i posteriorment les interaccions que hi podia dur a terme l'usuari.

I en aquest últim punt, la implementació, de la mateixa manera, hem parlat de la implementació d'aquests elements i després hem vist la corresponent a les interaccions.

Simplement això; recalcar aquesta separació que penso que ha estat útil i ens ha ajudat a anar comprenent els diferents punts de vista i, ja en una perspectiva final, a veure aquesta evolució. En els punts restants, parlarem de les conclusions finals que deixa la realització d'aquest projecte, així com les opcions de treball futur.

## **8.- CONCLUSIONS I ASSOLIMENT D'OBJECTIUS**

Deixem enrere la part central, el cos del projecte, i entrem en els apartats concloents d'aquesta memòria. I, per encetar aquesta part final, m'agradaria començar fent una valoració general i també d'aspectes més concrets sobre com s'ha desenvolupat el projecte i de si s'ha arribat o no als propòsits marcats al començament.



*Imatge 68: Conclusions*

### **8.1.- Valoració de la planificació i el desenvolupament**

El primer de què voldria parlar en aquest sentit és de la planificació i de la durada del projecte. No sé fins a quin punt ha resultat remarcable llegint l'apartat de planificació i desenvolupament però és un dels temes en què més penso que cal reflexionar i fer autocrítica.

Tal i com dic en el tercer punt, el desenvolupament temporal d'aquest projecte és atípic donades les pròrrogues i prolongacions que he fet. La primera penso que és força justificada; començo a treballar en una idea partint de zero (desenvolupament d'una aplicació pròpia) i al cap de poc temps es dona el fet que sóc seleccionat per dur a terme pràctiques en una empresa. D'aquesta manera, i tant per aprofitar per allargar aquesta estada com per no voler patir una càrrega de treball massa elevada decideixo allargar el projecte un quadrimestre més (fins a maig/2015 en comptes de febrer/2015).

No obstant, com ja comento en el tercer punt, cometo un error de planificació i sobretot de dedicació durant els primers mesos de 2015, la qual cosa provoca que hagi de tornar a posposar la presentació del projecte. Per tant, la primera conclusió que extrec de la meva experiència amb aquest projecte és més aviat negativa. Penso que aquesta segona pròrroga va ser conseqüència d'una mala conducta i espero poder convertir aquest error en una lliçó (i, per tant, en alguna cosa positiva) de cara al futur.

Dit això, és evident que vaig poder replantejar-me la planificació i penso que en aquesta segona part de 2015 ha estat bona. A més, penso que la comunicació amb els tres agents implicats (Tutor, Client i Desenvolupador) ha estat eficient i ordenada, per la qual cosa extrec en aquest sentit una conclusió bastant positiva d'aquest aspecte.

### **8.2.- Assoliment d'objectius i requeriments**

Un cop comentats aquests temes més generals del desenvolupament de la feina, entrem a valorar aspectes més tècnics del projecte en sí, valorant cadascun dels objectius establerts i analitzant el seu compliment o no, i fent el mateix també pels requeriments demanats per l'actor client.

#### **Objectius mòdul – usuari:**

*“1.- Obtenir una interfície intuïtiva. Que l'usuari compregui en tot moment què està fent i quins efectes causen les seves interaccions.”*

Penso que en aquest sentit la valoració és correcta, tot i que podria ser millorable. M'explico; aquest objectiu peca una mica de poca concreció, és a dir, sí que és valorable però parlem d'un aspecte que penso que sempre es pot millorar. Estic content del punt a què s'ha arribat però penso que mai ens hem de conformar, sobretot en aquest aspecte.

*“2.- Poblar les interfícies amb els elements justos i imprescindibles. No carregar l'espai amb imatges, textos o detalls innecessaris.”*

Aquest penso que és un objectiu que s'ha assolit bastant bé. Al llarg del projecte ha estat una premissa molt clara a seguir i hi ha hagut correccions i remodelacions en aquest sentit. Per exemple, en un primer moment la idea era que hi hagués sempre un element 'paperera', per a l'esborrat dels elements o de la frase, fixe en la pantalla, el qual va ser descartat justament per intentar assolir aquest objectiu.

*“3.- Donar suport visual amb animacions gràfiques. Sovint una petita animació pot ajudar a traslladar el significat de l'acció que acaba de tenir lloc en l'aplicació i alhora fa que sigui més amable visualment per a l'usuari. Important no excedir-se en l'elaboració en aquest aspecte tampoc.”*

Aquesta és una directriu que he tingut molt present al llarg del projecte i penso que és un dels aspectes a valorar de manera més positiva. Penso que he encertat bastant a l'hora d'incorporar animacions al mòdul i alhora he buscat (i penso que aconseguit) que aquestes fossin clares però sense adorns. Per això opino que és un dels objectius assolits de la millor manera.

## **Objectius mòdul – aplicació:**

*“1.- Acoblar el mòdul a l'aplicació de forma òptima. Estem parlant d'una aplicació desenvolupada en entorn professional, amb estructures de dades i elements de programació avançada. No és senzill aconseguir acoblar un mòdul aliè així com així, cal tenir clar com aquest interacciona amb l'aplicació pare.”*

Aquest, potser, és l'objectiu que em genera més recança de tots. En un principi, a l'hora de fixar els objectius, vaig pensar que en acabar el projecte, l'adaptació del mòdul a ECO seria instantània. No obstant, degut a la metodologia de desenvolupament escollida, hi ha un seguit de tasques addicionals abans d'aconseguir aquest acoblament òptim, les quals detallaré en el proper punt (treball futur).

*“2.- Respectar la interfície. És a dir, no allunyar-se de com l'aplicació ja definia el disseny, l'estructuració i comportament dels elements en les diferents pantalles. En aquest sentit, volem que de cara als usuaris sigui invisible aquest canvi (quan passin a la pantalla corresponent al mòdul del projecte).”*

Ben assolit. L'element més important de la interfície (la frase) utilitza una estructura visualment similar a d'altres en l'aplicació (l'exemple més clar, la llista de selecció de categoria de l'SlidingMenu) i també la reproducció segueix un patró casi idèntic al que ja hi havia. Sí que hi ha hagut alguna novetat o diferència (imatge paperera, imatges indicadores de contingut amagat, animacions afegiment o esborrat) però s'ha de tenir en compte que en alguns casos no hi havia cap precedent en l'aplicació i a més a més no hem d'oblidar que la pantalla per la qual hem desenvolupat el mòdul està dedicada a l'usuari final, i no a l'administrador (resta de pantalles).



*“3.- Respectar la programació. Estructurar i desenvolupar el mòdul de la manera més similar i acoblable possible a l'aplicació pare, fer ús dels elements i estructures que ja utilitzi l'aplicació. Important per a l'acoblament òptim del mòdul però també de cara a possible treball futur.”*

Content també en aquest sentit. Penso que en gran part he fet ús de mecanismes i elements ja presents en el codi de l'aplicació i els canvis que he incorporat han estat ampliacions de classes i components que ja hi havia i no pas desenvolupament d'altres de propis i específics per al meu mòdul.

#### **Requeriments:**

- *Col·locació i format de la frase:* Crec que tant la col·locació, com el format i la orientació són els òptims, força content.
- *Afegiments i insercions:* Penso que hem sabut implementar un mecanisme que, alhora que senzill, és molt clar.
- *Eliminacions:* Igual que l'afegiment i la inserció, i a més penso que haver incorporat el mateix mecanisme és un encert.
- *Reproducció:* Molt satisfet, sobretot amb la idea de deixar la frase destapada per a poder remarcar l'element que s'està reproduint en cada moment.
- *Animacions:* Amb animacions senzilles penso que hem aconseguit que les interaccions i els seus efectes siguin més intuïtius.
- *Indicadors de contingut:* És un petit detall, però penso que pot ser força útil (en alguns casos més que en d'altres).

Penso que en general la valoració d'objectius i requeriments és bona i que la fixació d'aquests (excepte potser en algun cas concret) va ser bastant realista i concreta. També penso que va faltar un punt de previsió d'imprevistos, en el sentit de pensar que les coses no sempre surten com s'han planejat. No obstant, reitero que quedo força content i convençut de la feina feta. Abans d'acabar, però, també m'agradaria valorar un parell d'aspectes més.

### **8.3.- Valoració personal**

En primer lloc vull dir que aquest projecte m'ha ajudat a valorar bastant el fet de aportar el teu gra de sorra a un castell ja construït. És a dir, en el moment que la concepció del projecte va canviar i va passar de ser de la realització d'una aplicació pròpia a un mòdul per una altra, reconec que vaig tenir un petit pensament de “tindré menys feina”. Per una banda aquesta afirmació era correcta, de ben segur que he hagut d'escriure menys línies de codi de les que hauria hagut d'escriure de l'altra manera, però per altra banda he pogut conèixer la feina que representa haver d'entendre el funcionament d'una aplicació professional i com entrar-hi.

Penso que és una tasca que potser no es valora gaire, però gràcies al desenvolupament d'aquest projecte he pogut comprendre la importància que té i la feina real que comporta. A més a més, penso que, com ja dic en la introducció, és un escenari que es pot donar més d'una vegada (i de dues) al llarg de la meua carrera, de

manera que espero tenir present aquesta idea en el futur, en el cas que es donin aquestes situacions.

Per últim, m'agradaria parlar del que al final és potser l'objectiu principal i més important de totes les tasques i feines que es desenvolupen, ja no tan sols al llarg d'una carrera, sinó d'una vida. Parlo, evidentment, de l'aprenentatge. En el cas d'aquest projecte, vaig passar també per un aprenentatge previ al desenvolupament d'aquest, de manera que potser per això penso que l'aprenentatge que m'emporto d'aquesta experiència és tan gran.

Valoro molt tot el que après durant aquests gairebé 2 anys en el camp del desenvolupament mòbil i concretament d'Android, sobretot si miro cap enrere i comparo els coneixements que tenia aleshores en la matèria i els que tinc ara. Simplement afegir que espero que aquest projecte hagi estat només una primera passa d'una exitosa carrera en el món del desenvolupament mòbil.

## **9.- TREBALL FUTUR**

Per anar acabant, tenim aquest interessant apartat al qual li voldria dedicar unes pàgines. Abans de veure la bibliografia i els annexos, m'agradaria parlar de les opcions de treball futur que es poden extreure de la meva aportació a l'aplicació ECO.

En un principi, típicament, aquest punt es dedica a proposar i descriure possibles millores a incorporar al sistema. No obstant, en el nostre cas, cal parlar primer d'un futur més immediat, ja que, com hem comentat en el desenvolupament del projecte, la manera que hem escollit per a treballar es seguir camins separats i, en acabat, dur a terme una unió manual.

Per tant, a continuació, detallaré aquesta situació així com les tasques que se'n desprenen, per a després ja sí concloure amb una valoració de possibles millores.

### **9.1.- Immediat**

Així doncs, la versió d'ECO sobre la qual he treballat, desenvolupat i implementat el meu mòdul és una versió que se'm proporciona el juliol de 2015 (després de la segona pròrroga del projecte, agafo una versió aleshores actualitzada per a tornar a començar a nivell de desenvolupament, abans havia treballat sobre una versió anterior).

Aquesta versió, com comentem en el desenvolupament, està pensada per a funcionar de manera òptima per a les versions 4.x d'Android, desenvolupada amb l'API 19 (Android 4.4.4). Per la seva banda, el professor desenvolupador segueix treballant i evolucionant l'aplicació, fins al punt que, actualment, està començant a fer proves per adaptar l'aplicació a les novetats introduïdes en l'Android 6, havent passat també per les de la versió 5.

Destaquem, això sí, que els canvis desenvolupats pel professor desenvolupador no entraran en conflicte amb les meves aportacions a nivell de funcionalitat. És a dir, per exemple, la disposició d'elements en la pantalla que hem modificat és la mateixa, amb les mateixes funcions, etc. El que cal ajustar són possibles detalls de la programació on sí que hi poden haver hagut variacions.

D'aquesta manera, primer de tot cal identificar els elements on, per ambdós camins, hi ha hagut canvis. Aquests són, per una banda, el **fragment** sobre el qual hem implementat la majoria d'ampliacions (CategoryFragment), i per l'altra, l'**adapter** on, com hem pogut veure en la implementació, en un principi podíem esperar no haver de fer-hi canvis però, sobretot per motius d'animació i altres detalls, hem hagut d'afegir-hi un seguit de *flags* per activar diferents modes i ampliacions.



*Imatge 69: Acoblament*

Així doncs, a partir d'aquesta situació, definim les passes a seguir i la distribució de la càrrega de feina per a l'òptim acoblament del mòdul en l'aplicació ECO:

- Per la meua banda, primer de tot, dur a terme un procés de **comentat** exhaustiu de tot el codi modificat tant en el fragment com en l'adapter.
- A continuació, és necessària una **reunió** amb el professor desenvolupador per tal d'assegurar la comprensió de tot aquest contingut. És a dir, repassar tota la implementació nova. Per sort, com que varies decisions d'implementació han

estat preses en comú, prou feina d'aquest punt ja estarà feta o, almenys, serà més lleugera.

- Finalment, ja tant per la meva com per la seva banda (professor desenvolupador), caldrà ajustar les **modificacions** pertinents tant en el meu mòdul com en el codi de la versió actualitzada d'ECO (la seva), per així aconseguir aquest acoblament òptim entre ambdues parts.

## 9.2.- Millores i propostes d'ampliació

Ara ja sí, deixant enrere aquest futur més immediat, parlarem de les possibles millores que, a causa d'haver treballat en un projecte determinat i haver-ne vist i ampliat el funcionament, hom pot extreure.

Evidentment, el primer a comentar en aquest punt és el tema de les validacions. Com ja comento en les conclusions, per temes de temps no ha estat possible incloure en el propi projecte un aspecte que ens hauria agradat, que és haver provat la nova interfície amb un usuari real. Penso que hauria estat interessant poder obtenir *feedback* en aquest sentit, i dur a terme les correccions pertinents (o si més no, descriure-les i incloure-les en aquest apartat).



Imatge 70: Millores

No obstant, com diem, això no ha estat possible per qüestions de temps, i tot i que tenia la confiança de treballar sota les condicions d'un actor client experimentat i coneixedor del tema, penso que el primer pas de la "nova" aplicació ECO hauria de ser el testeig amb usuaris reals.

El segon aspecte de què m'agradaria parlar és el de les icones o imatges. Totes les imatges i icones pel què fa a botons i imatges de les categories per defecte (com les que hem vist en les diferents captures d'aquesta memòria) han estat dissenyades per alumnes del Grau de Disseny de la UPC.

Aleshores, les que jo he aprofitat per a implementar els tres elements que necessitaven una imatge (capçalera de la frase i botons de reproduir i sortir de la frase) ja formen part de l'aplicació en algun altre context. Aquest escenari no és l'òptim, ja que, tot i ser icones adients a la seva funció, l'ideal és que cada funció tingui la seva icona.

D'aquesta manera, penso que un pas interessant seria crear les icones necessàries, en aquest cas, per aquests tres elements que comento. O, per altra banda, en cas d'una futura actualització de tot el conjunt d'icones d'ECO, tenir-los també en compte.

Per a veure les altres possibles millores o incorporacions que penso que serien interessants les enumero en el següent llistat:

- Una mica el que seria el pas següent al d'haver introduït la possibilitat de construir i reproduir una frase, penso que seria interessant poder **guardar** aquestes frases. És a dir, un cop construïda (i reproduïda, per què no), tenir la possibilitat de salvar en la base de dades aquesta construcció abans de procedir a eliminar-la, i més endavant, tenir la opció de recuperar-les.

Cal tenir en compte, no obstant, que això seria una opció a valorar per a l'usuari administrador, però que s'hauria de poder fer des de la pròpia pantalla de selecció de missatges. Això comporta, evidentment, incloure algun mètode o

interacció per a activar aquesta opció que fos una mica més complicada i poc probable que l'accionés l'usuari autista.

Evidentment, caldria incorporar també un control de contrasenya, igual com hi és quan s'accedeix al menú d'administració, ja que mai se sap si aquest usuari podria arribar a activar aquest mecanisme. És simplement una idea que penso, insisteixo, que seria el següent pas natural d'ECO.

- També en relació amb la frase, penso que podria ser interessant ampliar el repertori d'opcions que tenim un cop s'inicia la reproducció. Ara per ara, tenim la opció de sortir d'aquesta, però crec que podria aportar valor afegir-hi alguna opció més.

Penso, per exemple, en afegir la opció de parar la frase (la reproducció) i evidentment poder reprendre-la. Per a aquest propòsit podria ser interessant col·locar una espècie de barra de reproducció a la part inferior de la pantalla, de forma horitzontal, ja que en aquest cas necessitariem més espai per botons.

Ara, aprofitem per a sobreposar els dos que tenim (reproduir i sortir) ja que sabem que mai serà necessari que convisquin, però afegint les opcions que comento sí que caldria. Aleshores, si es creés una barra de reproducció també s'hi podrien afegir les opcions de 'missatge següent' i 'missatge anterior'. Una altra idea.

- Per acabar, un altre aspecte que podria ser interessant de valorar és la incorporació de noves interaccions per a dur a terme les accions d'afegiment/inserció i esborrat. Com ja he comentat, penso que són les més encertades en quant al seu caràcter intuïtiu, però en algun cas podria ser necessari tenir-ne d'altres.

Penso, per exemple, en el cas que a algun usuari li costés mantenir un objecte clicat o arrossegat-lo. Aleshores caldria veure quines altres interaccions incorporar. En aquest projecte, per exemple, es va valorar incorporar interaccions de doble clic; que en fer doble clic sobre un missatge de la graella aquest s'afegís al final i que en fer-ho en la frase, s'esborrés. No obstant, es va descartar precisament per què no resultava gaire intuïtiu.

Evidentment, en aquest cas s'hauria d'afegir en la part d'administració la opció de poder triar quina interacció es vol activar en aquesta pantalla.

## **10.- BIBLIOGRAFIA**

### **Material assignatures grau**

Material *DAMO* (Disseny d'Aplicacions Mòbils) – Curs 2013/2014, Q1

*Josep M. Merenciano*

Material *INDI* (Interacció i Disseny d'Interfícies) - Curs 2013/2014, Q1

*Bernardino Casas*

Material *INEP* (Introducció en l'Enginyeria del Programari) - Curs 2011/2012, Q1

*Josep M. Merenciano*

Material *AMEP* (Ampliació en l'Enginyeria del Programari) - Curs 2011/2012, Q2

*Josep M. Merenciano*

### **Web**

“Fragments” [Consulta Setembre/2015]

<http://developer.android.com/intl/es/guide/components/fragments.html>

“FrameLayout” [Consulta Setembre/2015]

<http://developer.android.com/intl/es/reference/android/widget/FrameLayout.html>

“LinearLayout” [Consulta Setembre/2015]

<http://developer.android.com/intl/es/guide/topics/ui/layout/linear.html>

“LinearLayout.LayoutParams” [Consulta Setembre/2015]

<http://developer.android.com/intl/es/reference/android/widget/LinearLayout.LayoutParams.html>

“ListView” [Consulta Octubre/2015]

<http://developer.android.com/intl/es/guide/topics/ui/layout/listview.html>

“ArrayAdapter” [Consulta Octubre/2015]

<http://developer.android.com/intl/es/reference/android/widget/ArrayAdapter.html>

“GridView” [Consulta Octubre/2015]

<http://developer.android.com/intl/es/guide/topics/ui/layout/gridview.html>

“*ImageView*” [Consulta Octubre/2015]

<http://developer.android.com/intl/es/reference/android/widget/ImageView.html>

“*AdapterView.OnItemLongClickListener*” [Consulta Novembre/2015]

<http://developer.android.com/intl/es/reference/android/widget/AdapterView.OnItemLongClickListener.html>

“*View.DragShadowBuilder*” [Consulta Novembre/2015]

<http://developer.android.com/intl/es/reference/android/view/View.DragShadowBuilder.html>

“*View.OnDragListener*” [Consulta Novembre/2015]

<http://developer.android.com/intl/es/reference/android/view/View.OnDragListener.html>

“*AnimatorSet*” [Consulta Desembre/2015]

<http://developer.android.com/intl/es/reference/android/animation/AnimatorSet.html>

